

IMPLEMENTING PURCHASING ASSISTANT USING PERSONAL PROFILE

Sanguk Noh

*School of Computer Science and Information Engineering
The Catholic University of Korea
Yeokgok 2dong, Pucheon-si, Republic of Korea
sunoh@catholic.ac.kr*

ABSTRACT

As a vast number of products have been flooding into the electronic commerce market, it would be almost impossible for customers to consider all the alternatives and make sensible choices. This problem motivates us to investigate the implementation of purchasing assistant for buyers in multi-agent trading settings. Towards this end, we formally define domain knowledge using ontology builder, and compile customers' preferences into personal profile. Based on personal profile, our recommendation agents prioritize the products according to their total utilities. In the experiment, it turns out that our recommendation agents properly advise a specific product based on the customer's preferences.

KEYWORDS

Recommendation Agent, Ontology, Personal Profile, User Preference, Multi-Attribute Utility Theory.

1. INTRODUCTION

A vast number of products and services have been flooding into the electronic commerce (e-commerce) market. It would be almost impossible for customers to consider all the alternatives and make sensible choices. If customers are tired of sorting out information needed for their purchase or they are not satisfied with their own resulting choices for their pains, they will not hesitate to leave the e-market. The suppliers also will not tolerate slow response from customers and uncertain orders which do not reflect customers' preferences faithfully. As the e-market is expanding all the more, its constituent members, that is, market providers, merchants, and customers will definitely require an assistant tool to make the e-market run in an active and efficient manner [6, 7, 9, 14].

This paper will focus on a purchasing assistant from customer's point of view. Our agent as a purchasing assistant recommends the feasible number of probable choices for customers using personal profiles in multi-agent trading settings. The role of our recommendation agent is to compile customers' preferences into their personal profiles, filter unprofitable choices based on the profiles, and hence recommend best choices among a huge number of products and services. This will lead to timely response between customers and merchants by reducing the range of choices, and more accurate purchasing decisions through analysis of customers' preferences.

As a case study, we implement a digital camera trading system as a multi-agent e-commerce market. The system will define domain knowledge in the form of ontology, which help communicate between the components of the system. Building the digital camera domain ontology, then, we test the quality of recommendation for the customers by computing the utilities of products given their preferences.

In the following section, we introduce an e-commerce trading system in multi-agent settings, and explain the components of the system. The ontological issues and customers' preferences will be discussed. In experiment, students as customers will evaluate digital cameras and their feedback will be stored in the form of their profiles. In conclusion, we summarize our results and further research issues.

2. MULTI-AGENT TRADING SYSTEM

To provide trading markets on the Internet, various trading environments have been suggested [6, 9, 14]. This paper examines the general skeleton of the trading markets, and introduces a multi-agent trading system which facilitates buyers and suppliers in the e-commerce market. The multi-agent trading system represents each role of the components, which are market provider, merchant agent, and recommendation agent, as depicted in figure 1. We implement the system using Java programming language so that the agents can be applicable regardless of any platforms.

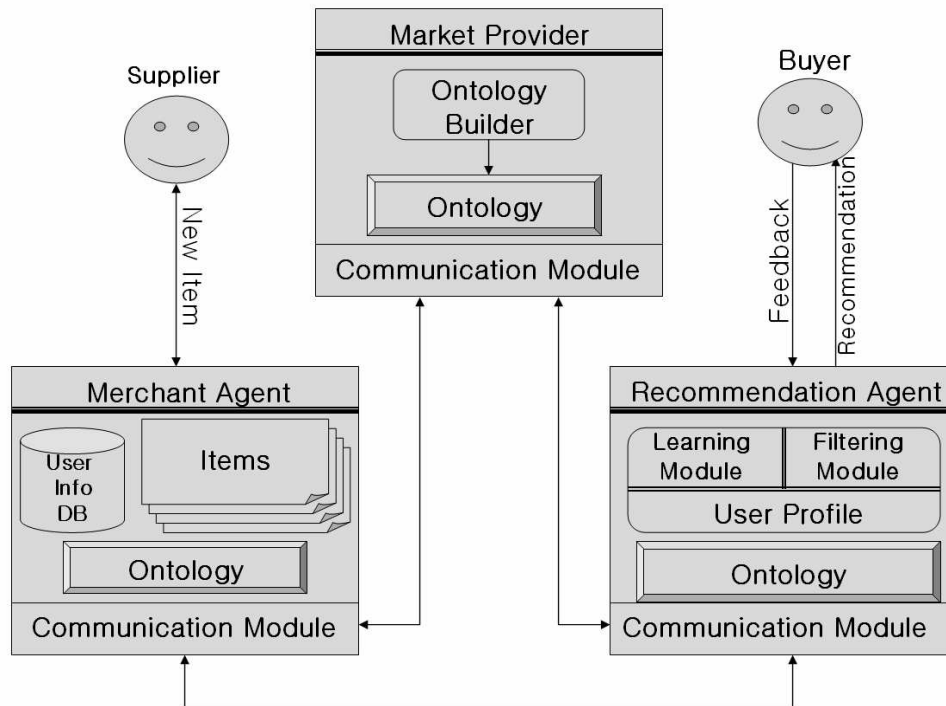


Figure 1. A multi-agent trading system architecture. The system consists of market provider, merchant agent, and recommendation agent.

The market provider module produces a specific ontology, using its ontology builder shown in figure 2, given an application domain. The generated ontology can be shared with merchant agent and recommendation agent through communication module of each agent, and enables the agents to interchange their knowledge [11, 13]. The market provider manages not only the ontology but also the information about the other agents. For example, to get the instantiations of products/items, the recommendation agent requests IP address of merchant agent from market provider. The peer-to-peer connection between the recommendation agent and the merchant agent can be established, and then the merchant agent provides the recommendation agent with the actual instantiations of products.

The merchant agent provides a graphic user interface (GUI) used to instantiate a tapestry of products from suppliers, and a list of products to recommendation agents, in which customers may be interested, given the ontology generated. The list of products and their instances can be passed over the recommendation agent whenever the recommendation agent requires.

As we mentioned, this paper focuses on the role of purchasing assistant, who prioritizes products based on personal profile. The recommendation agent as a purchasing assistant [2, 3, 7, 8] composes of learning module, filtering module, and communication module. The learning module of the recommendation agent compiles customers' preferences into user profile, and continuously updates the user profile to represent the up-to-date customers' preferences. Its filtering module recommends best products tailored to the customers' needs, based on their personal profile. Since the filtering module narrows down the number of alternatives far

from the customers' likeliness, it leads to practical choices among the huge number of products. The filtering process, thus, improves the responsiveness between the market and a specific customer, and reduces the ambiguity of the customer's decision.

2.1 Building an ontology

The ontology, in this paper, could be defined as the entity representing the hierarchy of information (or concepts) which various types of software agents communicate in multi-agent settings [1, 11, 13]. The ontology can interlink the market provider, merchant agent, and recommendation agent within the electronic trading system on the Internet, by providing terminologies and meanings. Thus, the ontology enables them to define their formal knowledge about the world, to share their knowledge, and to interchange their information.

As an ontology development tool, they have developed ONTOLINGUA [5], PROTÉGÉ-2000 [12], WebODE [1], etc. Duineveld et al. [4] and Noy and McGuinness [13] have surveyed their weaknesses and advantages for comparative studies. We have also designed and implemented an ontology builder using Java programming language, which easily define and edit attributes, and represent the hierarchical relation between the attributes.

The ontology in our framework hierarchically defines domain knowledge, and, from merchant agent's perspective, serves as a basic template of instantiation for a product and, from recommendation agent's perspective, serves as personal profile to represent customers' preferences. For example, the market provider defines a digital camera domain knowledge in form of ontology, as depicted in figure 2, using our ontology builder. 'Digital Camera' in figure 2 is the root class in the concepts of hierarchy. 'Spec' and 'CCD' are the objects, in the different level of the ontology, of the 'Digital Camera' class. The values of the leaves in the ontology, for example, 1 mega pixel, are the attributes in the form of slots.

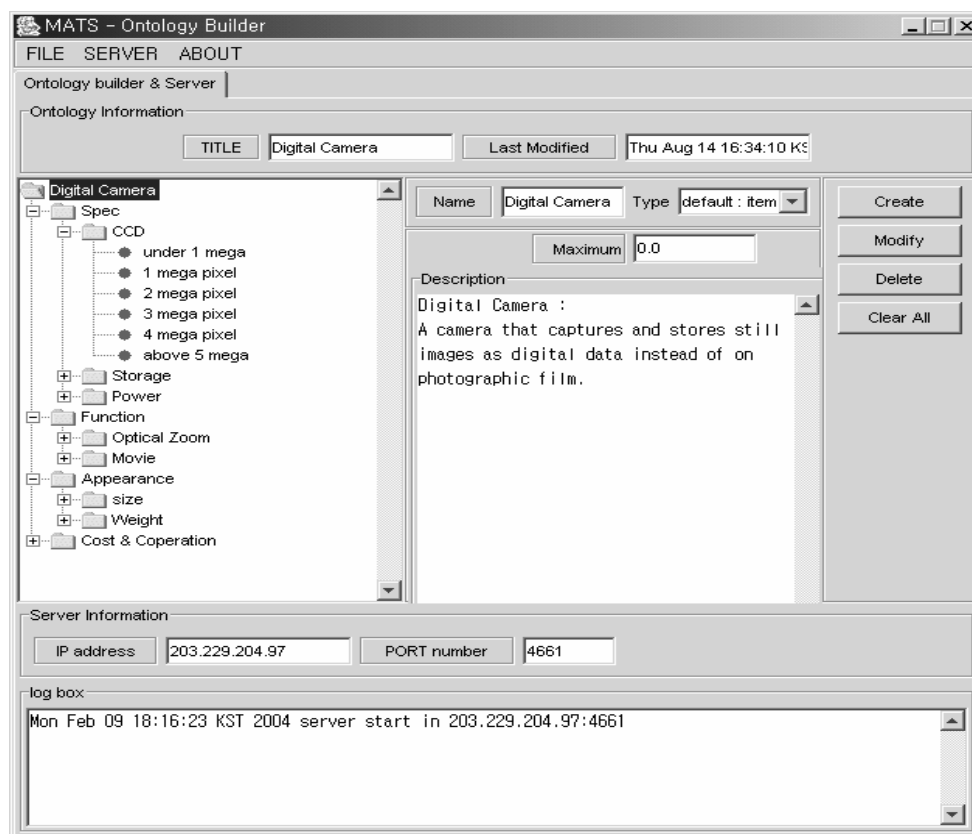


Figure 2. An example screen of ontology for digital camera. The digital camera domain ontology was generated by using our ontology building tool.

2.2 Calculating customers' preferences

Building domain ontology, the recommendation agent as a purchasing assistant maintains customers' preferences and prioritizes alternatives based on their preferences. To evaluate the preferences, our agents follow the multi-attribute utility theory [15, 16, 17, 18]. The utility represents user's preference or happiness. The total utility associated with a product can be summed up using weights of attributes. If a customer prefers a product, therefore, its utility could be greater than those of other products. The value or utility, V , of a product x at current time t can be computed as follows:

$$V(x|t) = \sum_{i=1}^n W_i(x|t) \quad (1)$$

where W_i is the weight of i -th attribute. Equation 1 is a function evaluating the value of a product, which presents customer's preference.

The customers refine their preferences and make purchasing decisions. To maintain up-to-date customer's preferences, the recommendation agent gets their feedback and updates the weights of attributes in their personal profile as follows:

$$W_i(x|t) = W_i(x|t-1) + \Delta W \quad (2)$$

In equation 2, when the customer replies with 'good,' ΔW is given to 0.5; otherwise, ΔW is 0.

Table 1. An example of customer's feedback in a digital camera domain

	Pixels CCD	Optical Zoom	Manufacturer	Evaluation
Camera A	2 MP	3×	Sony	No Good
Camera B	3 MP	5×	Samsung	Good
Camera C	2 MP	None	Samsung	No Good
Camera D	3 MP	5×	Canon	No Good

For example, let us consider table 1. Table 1 contains four digital cameras from Camera A to Camera D. When a customer evaluates them as shown in table 1, the weights of the attributes of Camera B rated with 'good' increase by 0.5. According to equations 1 and 2, the resulting utilities of the four digital cameras should be changed as follows:

$$\begin{aligned} V(A|t) &= 0 + 0 + 0 = 0 \\ V(B|t) &= 0.5 + 0.5 + 0.5 = 1.5 \\ V(C|t) &= 0 + 0 + 0.5 = 0.5 \\ V(D|t) &= 0.5 + 0.5 + 0 = 1.0 \end{aligned}$$

As a result, the utility of camera B, $V(B|t) = 1.5$, is the greatest value among those of four digital cameras, and the camera B could be the best product considering customer's preference.

3. IMPLEMENTATION AND EVALUATION

We have implemented the multi-agent trading system¹ using Java programming language. To evaluate our purchasing assistant within the multi-agent trading system, we tested the system with 20 students who took

¹ For demonstration of the multi-agent trading system, refer to "http://cis.catholic.ac.kr/."

the computer seminar class. The goal of the experiment is to assess the quality of recommendation tailored to customers' personal preferences.

For the experiment, we used a digital camera domain. First of all, the market provider defined a digital camera domain ontology, as depicted in figure 2, and the merchant agent provided a list of products using the ontology, as shown in figure 3. The recommendation agent provided customers with the instantiations of the products, as captured in figure 4, and, if the customers set their preferences, then the recommendation agent compiled the preferences into their personal profile. The customer could refer to the resulting ranks on the digital cameras, as depicted in the right-most table of figure 4. This mapping process from the user preference to the alternative products enabled the customers to choose best products given a tapestry of products.

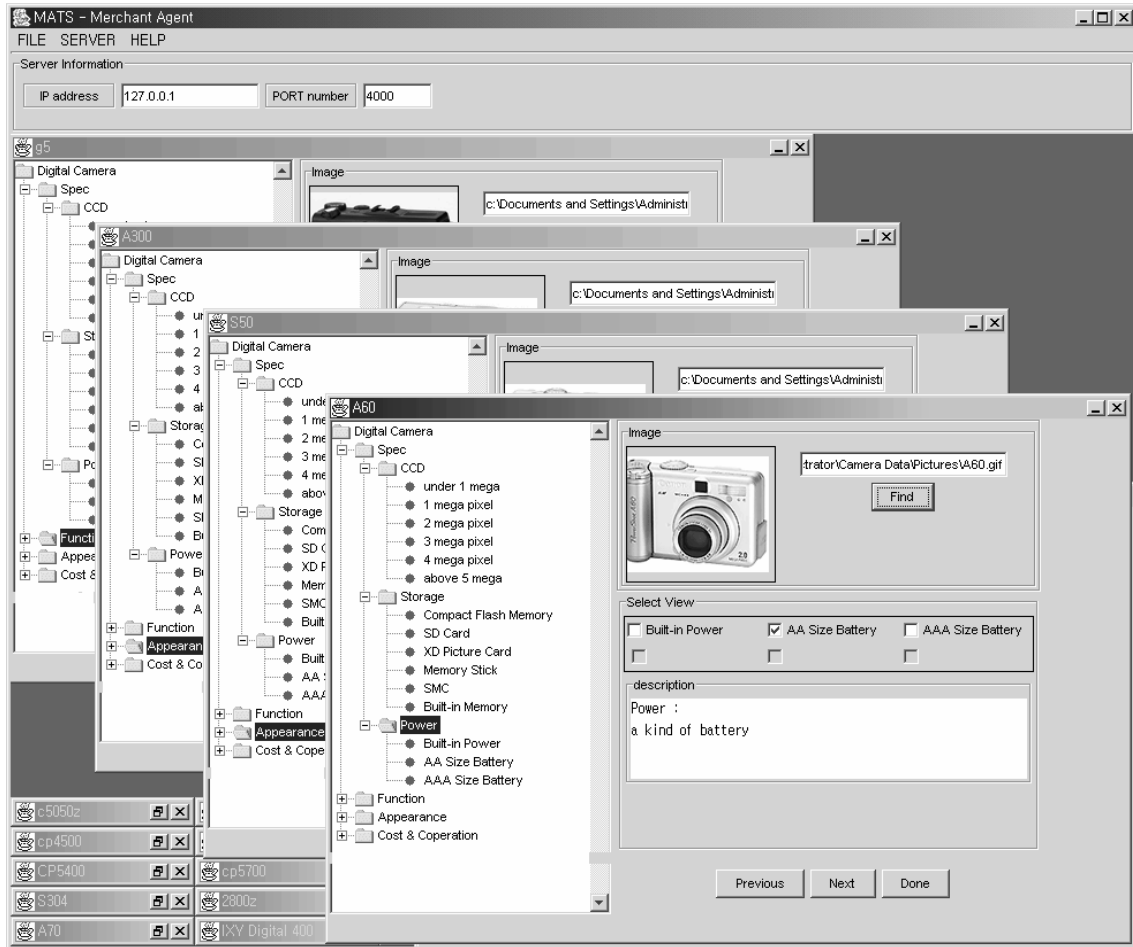


Figure 3. The screen of merchant agent shows a set of products provided by suppliers.

In the experiment, the merchant agent provided 100 digital cameras, as shown in figure 3, and, given the digital cameras, the customer expressed their preferences of the products by choosing 'good' or 'no good,' as shown in figure 4. Using equations 1 and 2 in Section 2.1, we measured the weights of attributes updated through the customer's feedback, and normalized the weights between 0 and 1.

We also computed the posterior probabilities of attributes using naïve Bayesian classifier given that the customer evaluated products as 'good.' Bayes rule examines whether or not a property² observed as evidence belongs to a specific hypothesis (or class), given a set of data distribution. Bayes theorem [10, 16] can be defined as follows:

² In this paper, the property corresponds to customer's preference or propensity.

$$P(h_j | x_i) = \frac{P(x_i | h_j)P(h_j)}{\sum_{j=1}^m P(x_i | h_j)P(h_j)}$$

where, a set of observable attributes, $X = \{x_1, x_2, \dots, x_n\}$; a set of hypotheses in a domain, $H = \{h_1, h_2, \dots, h_m\}$; $P(h_j|x_i)$ is the posterior probability of the hypothesis h_j , $h_j \in H$, given that x_i , $x_i \in X$, is an observable event. In our experiment, the set of observable attributes, X , consists of specification, function, appearance, and manufacturer of digital camera, and the hypotheses are either 'good' or 'no good.' Given the set of data as evidence naïve Bayesian classifier allows us to calculate the buyer's preferences, $P(x_i|h_j)P(h_j)$.

MATS - Recommendation Agent

User Preference Recommendation Agent

Name: S45
Price: 48.0

Description:
Digital Camera :
a camera that captures and stores still images as digital data instead of on photographic film.

Attribute:
 CCD : 4 mega pixel
 Storage : Compact Flash Memory
 Power : Built-in Power
 Optical Zoom : 2-3
 Movie : Include
 size : Medium
 Weight : 259.0
 Cost : 48.0
 Manufacturer : Canon

Item List

Name	Company	Price
DSC-D770	Sony	57.0
DSC F505v	Sony	78.0
DSC P30	Sony	20.0
DSC-P3	Sony	37.0
FD 200	Sony	71.0
DSC P71	Sony	47.0
Syber Sh...	Sony	28.0
DSC-P92	Sony	56.0
SDC 85	Samsung	14.0
Digimax 8...	Samsung	15.0
SDC-007	Samsung	18.0
Digimax 2...	Samsung	20.0
Digimax 2...	Samsung	28.0
Digimax 4...	Samsung	36.0
SCD-5000	Samsung	130.0
SDC-100	Samsung	11.0
X-2	Olimpus	54.0
C-2z	Olimpus	21.0
C-2	Olimpus	17.0
E-20	Olimpus	134.0
C4000Z	Olimpus	51.0
u 400 Digi...	Olimpus	49.0
C-960Z	Olimpus	18.0
C-840L	Olimpus	17.0
CP4300	Nikon	51.0

Recommendable Item List

Rank	Name	Score
0	S45	38.5
1	350se	38
2	M603	37
3	Digimax 410	36
4	cp3500	36
5	S50	36
6	CP885	35.5
7	CP880	35.5
8	v3	35.5
9	A70	35
10	X-2	34.5
11	F601Z	34.5
12	p7	34.5
13	digimax 330	34.5
14	mu 300	34.5

Choose Item in the list that you want.

Search items Good No Good

Figure 4. The recommendation agent provides a list of products instantiated. Customers set their personal preferences, and the recommendation agent, then, prioritizes the products based on the profile, as shown in the right-most table.

Figure 5 shows the preferences of one of 20 students. In figure 5, the white bar presents the weights measured using the weight updating rule, as described in equation 2, and the black bar presents the conditional probabilities measured using naïve Bayesian classifier. In this example, the student evaluated 18 alternatives as 'good' out of 96 digital cameras. To summarize the student's preferences in figure 5, the customer preferred (1) above 5 mega pixels CCD, (2) CF memory, (3) built-in battery, (4) $4 \times \sim 5 \times$ optical zoom, (5) voice included, and (6) large size. Most of the preferences measured using weights and probabilities were similar. We summarized all of the students' preferences using weight updating rule and naïve Bayesian classifier, and the resulting preferences were similarly represented by two methods. Thus, it turns out that our recommendation agents properly advised a specific product based on the customers' preferences.

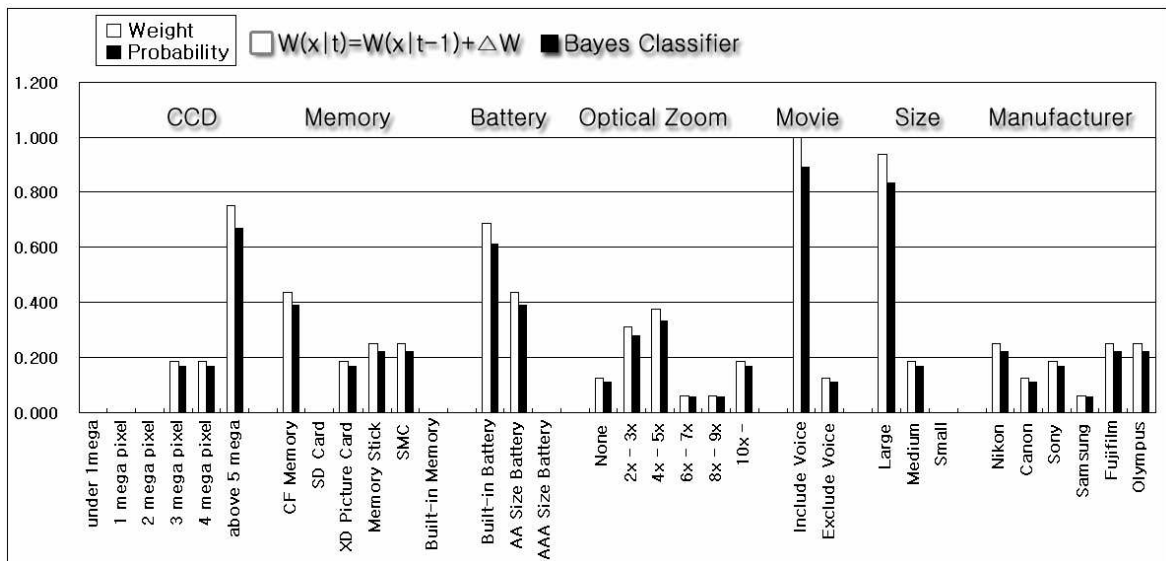


Figure 5. The resulting preferences of a specific customer. The white bar presents the weights measured using the weight updating rule, as described in equation 2, and the black bar presents the probabilities measured using naïve Bayesian classifier. For comparison, the weights are normalized between 0 and 1.

4. CONCLUSIONS

This paper implemented a plausible trading environment in multi-agent settings and focused on a question of how customers could get best products based on their personal preferences. Towards this end, we formally defined domain knowledge using ontology builder, and compiled customers' preferences into personal profile in a digital camera domain. Based on personal profile, our recommendation agents prioritized the products according to their total utilities. We compared the resulting weights of the attributes with the resulting posterior probabilities using naïve Bayesian classifier. The experiment showed that the customers within the electronic market were satisfied with the quality of recommendation.

Our next step is to improve our system to reflect customers' preferences in more refined manner. For example, the feedback could be classified by detailed levels, instead of 'good' or 'no good.' The recommendation agent, thus, interprets the strength or the weakness of a certain product, and quantifies the evaluation into some numbers. Further, we could extend our framework to other various e-commerce applications, for example, travel guidance, mobile contents, and so on. As well as buyer's perspective in this paper, we will design and implement trustiness and reputation of suppliers. Within our e-society, the merchant agent can keep track of suppliers' behaviors, and maintain trustworthy suppliers. By adding these functions, we hope we will develop a comprehensive multi-agent trading environment on the Internet.

ACKNOWLEDGEMENT

This work has been supported by the Korea Research Foundation Grant KRF-2003-041-D20465. We would like to thank our students, Beomsoo Cho and Jaewon Kim for their help in implementing and experimenting with the multi-agent trading system in a digital camera domain.

REFERENCES

- [1] Arpirez, J. C. et al., 2003. WebODE in a Nutshell. *AI magazine*, Vol. 24, No.3, pp. 37-47.
- [2] Bichler, M., Kalagnanam, J., and Lee, H. S., 2002. RECO: Representation and evaluation of configurable offers. In H. K. Bhargava and N. Ye (Eds.): *Computational Modeling and Problem Solving in the Networked World: Interfaces in Computing and Optimization*, Kluwer Academic Publishers.
- [3] Cosley, D. et al., 2003. Is seeing believing?: how recommender system interfaces affect users' opinions. *Recommender systems and social computing*, pp. 585-592.
- [4] Duineveld, A. et al., 1999. Wonder Tools? A Comparative Study of Ontological Engineering Tools. In *Proceedings of the Twelfth Workshop on Knowledge Acquisition, Modeling, and Management*, Banff, Canada, pp. 16-21.
- [5] Farquhar, A. et al., 1997. The ONTOLINGUA Server: A Tool for Collaborative Ontology Construction. *International Journal of Human Computer Studies*, 46(6), pp. 707-727.
- [6] Guttman, R., Moukas, A., and Maes, P., 1998. Agent-mediated electronic commerce: A survey. *Knowledge Engineering Review*, 13(2), pp. 147-159.
- [7] Huhns, M.N., and Singh, M.P., September/October 1998. Personal Assistants. *IEEE Internet Computing*, pp. 90-92.
- [8] Kitamura, Y., Sakamoto, T., and Tatsumi, S., 2002. A Competitive Information Recommendation System and Its Behavior. In M. Klusch, S. Ossowski, and O. Shehory (Eds.): *Cooperative Information Agents VI, Lecture Notes in Artificial Intelligence 2446*, Springer-Verlag, pp. 138-151.
- [9] Menczer, F. et al., July 2002. IntelliShopper: A Proactive, Personal, Private Shopping Assistant. In *Proceedings of International Conference on AAMAS*, Bologna, Italy, pp. 1001-1008.
- [10] Noh, S. and Gmytrasiewicz, P. J., 1999. Towards Flexible Multi-Agent Decision-Making Under Time Pressure. In *Proceedings of International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, pp. 492-498.
- [11] Noh, S. et al., October 2003. Classifying Web Pages Using Adaptive Ontology. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Washington, D.C., U.S.A., pp. 2144-2149.
- [12] Noy, N. F., Fergerson, R. W., and Musen, M. A., October 2000. The Knowledge Model of PROTÉGÉ-2000: Combining Interoperability and Flexibility. In *Proceedings of the Twelfth International Conference in Knowledge Engineering and Knowledge Management (EKAW-00)*, Juan-Les-Pins, France.
- [13] Noy, N. F., and McGuinness, D. L., 2001. Ontology Development 101: A Guide to Creating Your First Ontology. *Stanford Technical Report*, TR KSL-01-05.
- [14] Owen, M. J. et al., July 1999. Multi-Agent Trading Environment. *BT Technology Journal*, Volume 17, No 3, pp. 33-43.
- [15] Roth, R., Field, F., and Clark, J. Multi-Attribute Utility Analysis. In online overview at <http://msl1.mit.edu/>.
- [16] Russell, S. and Norvig, P., 2002. *Artificial Intelligence: A Modern Approach. Chapters 13 and 16*. 2nd edition, Prentice Hall, U.S.A.
- [17] Schaefer, R., 2001. Rules for Using Multi-Attribute Utility Theory for Estimating a User's Interests. In *Proceedings of the 9th GI-Workshop*.
- [18] Schmitt, C., Dengler, D., and Bauer, M., 2002. The MAUT-Machine: An Adaptive Recommender System, In online proceedings of ABIS workshop at http://www.kbs.uni-hannover.de/~henze/lla02/abis_proceedings.html.