Tri-Training based Bilateral Multi-Issue Negotiation Framework

Zhijun Zheng and Yanbin Peng
School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Hangzhou, China
Email: zhengzhijun@zust.edu.cn

***Abstract*—Negotiation has been extensively discussed in electronic commerce for decades. Recent growing interest in importing machine learning algorithm in electronic commerce has given increased importance to automated negotiation. A Tri-Training based algorithm was proposed to learn opponent's negotiation preference. The process of negotiation was viewed as a proposal's sequence which can be mapped into bidding trajectory feature space to form sample set. Due to fierce competition, unlabeled training examples are readily available but labeled ones are fairly expensive to obtain. Therefore, Tri-Training, as a semi-supervised method, was imported into negotiation framework to increase the number of samples and improve perdition accuracy of opponent's negotiation preference learning. Based on negotiation preference of both side, an optimization algorithm is conducted to compute win-win counter proposal. The experimental results show that the proposed method can decrease the number of negotiation steps and increase the overall utility of negotiation.**

***Index Terms*—negotiation framework; negotiation preference; semi-supervised; optimization**

I. INTRODUCTION

Along with the rapid development of agent technology, it has been used in e-commence widely. Agent can handle complex business activities efficiently instead of human. Automated negotiation is the core part in e-commence. Therefore, how to combine agent technology into automated negotiation is a challenge problem. In agent-based automated negotiation, there are three aspects which needs to research: negotiation protocol, negotiation content and negotiation policy. Among which, negotiation policy is the most important research point. Negotiation policy indicate negotiation agent how to response to the proposal of counter-partner. This paper convert negotiation problem into a machine learning problem. Firstly, negotiation preference is extracted from data of history negotiation information, which can be used to generate negotiation counter-proposal. Secondly, machine learning algorithm is used to predict negotiation preference of opponent.

In agent-mediated e-commerce [1], a key challenge is the way in which the agents negotiate to establish contracts with one another to provide particular products. In many cases, agent not only bargain over price of products, but also take into account aspects like quality, delivery time, warranty and payment method. This is so called multi-issue negotiation. In this setting, different agents have different preference attached to the different issues. Therefore, it is often possible to reach an agreement that is mutually beneficial for both parties. Due to fierce competition, agent prone to conceal their preferences, utility functions or reservation values for fear of being cheated. This hampers the mentioned win-win scenario. Therefore, the best solution is to approximate these preferences based on historical negotiation data and generate counter-offers based on preferences of both parties.

Recently, there are a lot of literatures [2-8, 14-16] which introduce machine learning methods to solve the negotiation problem. Sun [14] aims to help negotiation agent to select its best actions and reach its final goal, he proposed a bilateral price negotiation strategy based on Bayesian classification and Q-learning. Wherein, Bayesian learning and Q-learning are primitive form of learning. In the middle of negotiation process, negotiation agent makes the best use of the opponent's negotiation history to make a decision of the opponent's classification based on Bayesian classification, dynamically adjust the negotiation agent's belief of opponent in time, quicken the negotiation result convergence and reach the better negotiation result by Q-learning. Peled [15] considers many negotiations in the real world are characterized by incomplete information, and participants' success depends on their ability to reveal information in a way that facilitates agreement without compromising the individual gains of agents. He presents a novel agent design for repeated negotiation in incomplete information settings that learns to reveal information strategically during the negotiation process. The agent used classical machine learning techniques to predict how people make and respond to offers during the negotiation, how they reveal information and their response to potential revelation actions by the agent. The agent was evaluated empirically in an extensive empirical study spanning hundreds of human subjects. Ng [16] discusses the implement of machine learning approach in negotiation agents that can learn their opponent's preferences and constraints during one-to-many negotiations. A novel mechanism in learning negotiation is introduced in this paper. The genetic-based model of multi-attribute one-to-many negotiation, namely GA Improved-ITA is proposed. The GA Improved-ITA agents first utilize genetic-based machine learning to identify their opponent's preferable

© 2014 ACADEMY PUBLISHER
doi:10.4304/jsw.9.5.1129-1134

negotiation issues. It is then followed by branch and bound search to search for the best value for each of the issues. Cheng [9] labels the negotiation sample automatically by making full use of the implicit information in negotiation process. Afterwards, the labeled data become the training samples of support vector regression machine which outputs the estimation of opponent's utility. Finally, the utility of both sides constitute a constraint optimization problem which be figured out by genetic algorithm. The optimal solution is the counter-proposal. Coehoorn [10] try to learn negotiation preference with respect to the provision of a particular service. The particular approach Coehoorn used is kernel density estimation (KDE for short) which is a statistical method known to provide a simple way of finding structure in data sets without the imposition of a parametric model. Coehoorn explores and evaluates the use of KDE for negotiation preference learning and show how this approach can make the negotiation outcome more efficient for both participants. Hindriks [11] considers that the efficiency of automated multi-issue negotiation depends mainly on the availability and quality of knowledge about the negotiation opponent. Hindriks presents a generic framework based on Bayesian learning to learn the negotiation opponent model, including the issue preferences and the issue priorities of opponent. Hindriks's algorithm can effectively learn opponent preferences from proposal exchanges by making some assumptions about the preference structure and rationality of the negotiation process. Experimental results demonstrated the effectiveness of Hindriks's approach. Cheng [12] proposed a support vector machine based method to learn opponent's preference. In this framework, the process of negotiation was viewed as a proposals' sequence which can be transformed to multiple negotiation tracks by mapping them to a new feature space. Opponent's preference of each issue can be gained by learning the negotiation tracks. At last, a negotiation decision making model was used to calculate win-win counter-proposal.

Most of the current negotiation model assuming that there are a large number of labeled sample points, which is not in line with the actual situation. Hindriks uses Bayesian approach to estimate negotiation parameters. Although it is an unsupervised method, but as posteriori model, there is too many parameters to be estimated. The results of approximate calculation method are not precise enough. In the e-commerce environment, due to fierce competition, negotiation agent usually will not take the initiative to publish private information (including the current negotiation parameters and historical negotiation parameters), unlabeled training examples are readily available but labeled ones are fairly expensive to obtain. Therefore, supervised machine learning methods are not suitable for solving the problem of automated negotiation.

In recent years, semi-supervised learning has received more and more attention in machine learning research field. The intention behind semi-supervised learning is to import a large number of unlabeled sample points to help build a better classifier with more powerful prediction ability. For many real-world applications, unlabeled data are abundant but labeled data needs expensive human labor and time. Take web pages classification for example, it is convenient to collect web pages from internet, but labeling those web pages into different topics may require domain expert to work for a long time. Therefore, it is important to make use of unlabeled data to improve the prediction ability of classifiers. As time went by, semi-supervised learning becomes a hot research direction in machine learning.

Based on our former research works [4-8], this paper proposes a tri-training [13] based negotiation preference learning method. In this method, the process of negotiation is viewed as a proposal's sequence which can be mapped into bidding trajectory feature space. Against the problem of insufficient labeled samples, three machine learner iterations reliably provide labeled samples for each other, in order to expand the size of the training sample. Then, three learners get the opponent's negotiation preferences together. Ultimately, the negotiation decision model proposed a win-win negotiation counter proposal based on negotiation preference of both side. The experimental results show that the new method can improve the overall utility, reduce the number of negotiation rounds, and save negotiation time.

This paper is organized as follows. In section 2.1 we formalize our negotiation framework. Section 2.2 presents our classification scheme for opponents' preferences. Section 2.3 give an optimal algorithm for calculating counter-proposal, and finally sections 3 and 4 offer experiments and conclusions.

II. NEGOTIATION FRAMEWORKS

Three distinguished work have been proposed in this section. Section 2.1 presents the formal model which describes the negotiation. Section 2.2 describes the tri-training algorithm we use for predicting the negotiation preference. Section 2.3 presents the method for calculating the counter proposal.

A. Formal Description of Negotiation Model

A negotiation model can be modeled by a 9-tuple *<A,I,S,R,P,Utility,W,History,Protocol>*, where ,

A: the set of negotiating agents. $a_j \in$ A represents a specific negotiating agents (j∈ {1,2}).

I: the set of issues. $i \in I$ represents the issue under negotiation, such as price and quality.

S: the set of domains. For each issue *i*, every agent has a lower and an upper reservation value, resulting in a domain s_i=[min_i,max_i]. These values represent the best and worst value still acceptable for the agent. The value outside the domain is unacceptable for the agent.

R: An integer represents the number of round in negotiation process.

P: the proposal of negotiating agent. proposal P={x_1,x_2,…,x_n} is a set of values for all issues. x_i is value of issue *i*.

© 2014 ACADEMY PUBLISHER

Utility: To evaluate the value of an issue, each agent has a scoring function over its issue domain: *utility*: $s_i \longrightarrow [0,1]$ which assigns a score for every issue value x.

W: each agent has a weight vector over the issues, representing the relative importance of its issues. $w_i \in W$ is the weight of issue *i* of the agent. These weights should be normalized $\sum w_i = 1$. Therefore, the utility of proposal p can be defined as: $u(p) = \sum w_i \cdot utility(x_i)$.

History: history data of negotiation. All negotiation proposals are stored in negotiation database. Training samples of machine learning algorithm is obtained from history data.

Protocol: the negotiation protocol is alternating offer protocol. Formally, let $b^t_{a1\to a2}$ be the proposal of agent a1 to a2 at time t and $b^t_{i,a1\to a2}$ denote the value of issue i of this proposal. In our model we use discrete time. The agent who sends the first proposal is chosen at random. After the first proposal, the two agents send counter-proposal alternatively. At last, negotiation ends with two special proposals: {accept, refuse}.

Definition 1. Negotiation process is expressed as follows: $b^1_{a1\to a2}$, $b^1_{a2\to a1}$, $b^2_{a1\to a2}$, $b^2_{a2\to a1}$,…, $b^n_{a1\to a2}$, $b^n_{a2\to a1}$; $a1, a2 \in C$. Wherein $b^i_{a1\to a2}$, $b^i_{a2\to a1}$ is the ith negotiation round.

Definition 2. Given agent and its associated utility function, the counter-proposal of agent a1 at time t of a proposal $U(b^{t-1}_{a2\to a1})$ sent at time t-1 is defined as:

$$b^t_{a1\to a2} = \begin{cases} refuse & if\ U(b^{t-1}_{a2\to a1}) < U_{\min} \\ accept & if\ U(b^{t-1}_{a2\to a1}) \geq U(b^t_{a1\to a2}) \\ b^t_{a1\to a2} & else \end{cases}$$

Agent a1 decides which of the alternatives to choose from definition 2. When the utility of proposal of agent a2 is lower than tolerance's value (minimum value which is acceptable) of agent a1, agent a1 will refuse to continue negotiation. If this proposal has a higher utility than the proposal of agent a1 itself, agent a1 will accept it. In these two cases, negotiation process ends. Otherwise, the calculated counter-proposal will be made.

Definition 3. let $b^{j+1}_{i,a1\to a2}$, $b^{j+2}_{i,a1\to a2}$,…, $b^{j+k}_{i,a1\to a2}$ be the k-length issue sequence of agent a1 on issue i. let $\tilde{b} = max(b^{j+1}_{i,a1\to a2}, b^{j+2}_{i,a1\to a2}, ..., b^{j+k}_{i,a1\to a2})$

$$z^{j+1}_{i,a1\to a2} = (b^{j+1}_{i,a1\to a2} - \tilde{b})/\tilde{b}$$

$$z^{j+2}_{i,a1\to a2} = (b^{j+2}_{i,a1\to a2} - \tilde{b})/\tilde{b}$$

$$\cdots$$

$$z^{j+k}_{i,a1\to a2} = (b^{j+k}_{i,a1\to a2} - \tilde{b})/\tilde{b}$$

Then $z^{j+1}_{i,a1\to a2}$, $z^{j+2}_{i,a1\to a2}$,… $z^{j+k}_{i,a1\to a2}$ is the k-length bid trajectory of agent a1 on issue i.

Bid trajectory curve reflects changes in the amplitude of the concessions, concession amplitude changes reflects negotiation preferences of negotiation agent on issues. The faster the concession amplitude changed, the lower the agent's preference, as shown in Fig. 1. It needs to be noted is that the "bid" is a broad statement, including price, quality, warranty time etc.

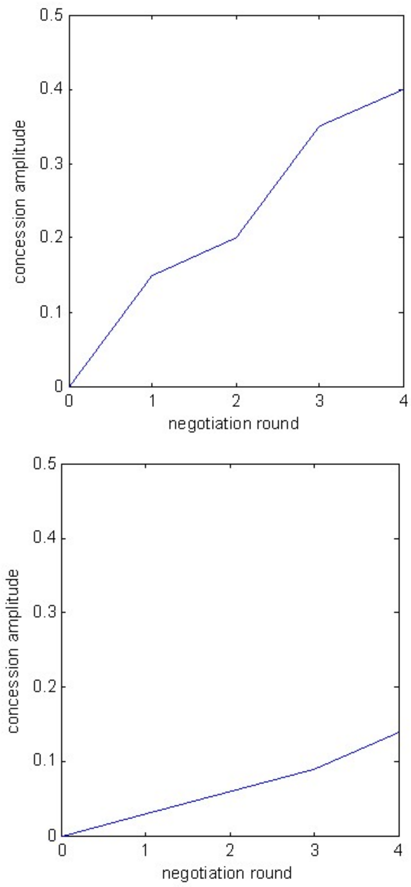

Figure 1. Bid trajectory (top: fast concession, small preference; bottom: slow concession, large preference).

Negotiation decision process is shown in Fig. 2. Negotiation opponent sends a negotiation proposal, which is received and stored in negotiation history database. In negotiation history database, we extract training samples and current negotiation feature (bid trajectory). Tri-training learner is used to predict current preference of negotiation opponent. Particle Swarm Optimization algorithm (PSO) is used to calculate win-win counter-proposal based on preference of both sides. At last, negotiation decision module makes the decision.

© 2014 ACADEMY PUBLISHER

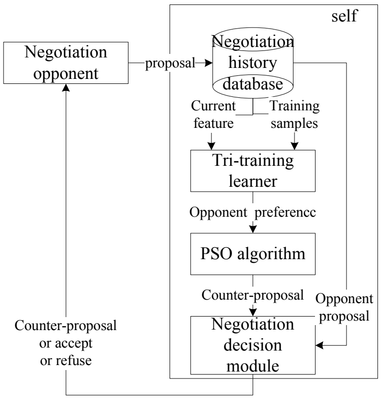

Figure 2. Negotiation decision process.

B. Tri-training Learner

Our goal is to introduce a learning method which can be used to calculate opponent's preference in a negotiation process. In this scene, negotiation process can be views as a trajectory of bidding. We exact bid trajectory from negotiation history to form training samples, which is used to train tri-training learner. Then, the trained learner is used to calculate opponent's preference based on the current bid trajectory of opponent.

In many practical machine learning applications such as face classification, unlabeled training examples are readily available but labeled ones are fairly expensive to obtain. Hence, semi-supervised learning algorithms such as tri-training have attracted much attention. This algorithm generates three classifiers from original labeled examples. These classifiers are refined in several rounds. In each round, an unlabeled example is labeled for a classifier if the other two classifiers agree on the label. The pseudo code of tri-training is as follows:

```
Tri-training(L,U,SVR)
Input: L: original labeled sample set
U: unlabeled sample set
SVR: support vector regression machine
For i = 1:3
    Train SVR_i in L;
    L_i = L;//labeled sample set of SVR_i
    U_i = U;//unlabeled sample set of SVR_i
End for
Repeat until none of L_i changes
        For i = 1:3
                For every x∈ U_i
                If SVR_j(x)=SVR_k(x) (j,k≠ i)
                        L_i=L_i ∪ {x};
                        U_i=U_i-{x};
                End if
        End for
    For i = 1:3
        ReTrain SVR_i in L_i;
    End for
End repeat
```

Output: $SVR_{tri}(x)= \arg\max_{y\in labelset} \sum_{i:SVR_i(x)=y} 1$

C. Negotiation Counter-proposal Generation

After we get opponent's preference $P=(t_1,t_2,\ldots,t_n)$, wherein t_i stands for preference of issue i. the optimal counter-proposal should first meet their own utility, and at the same time satisfy opponent's utility. Counter proposal is expressed as $b=(b_1,b_2,\ldots,b_n)$, optimal counter proposal expressed as b^*. Union utility function is expressed as $ALL(b) = \lambda MY(b)+(1-\lambda)OPPO(b)$, wherein function $MY(b)$ is our utility function; $OPPO(b)$ is negotiation opponent's utility function; λ is a weight used to compromise between our utility and opponent's utility, λ = 1 represents only our utility is considered, λ = 0 represents only opponent's utility is considered. According to the above definition, the optimal counter proposal can be expressed as: $b^* = \arg\max_{b\in\Gamma} ALL(b)$, wherein Γ stands for Hypothesis space. We use PSO algorithm [17-19] to solve the optimization problem.

III. Experimental Results

In order to validate the effect of the proposed negotiation model, we do some experiments in multi-agent based e-commerce platform developed in the laboratory. In the experiment, an electric fan trading scenario is simulated. This trade starts when the buyer agent sends out a proposal for electric fan to a seller agent. The seller agent search for price and other parameters of electric fan in their database and, in return generate a counter offer to the buyer agent. The buyer and the seller agents' motivation is to make more profit. There are two indicators which mainly reflect the performance of negotiation model: 1) negotiation round number, which refer how much round the negotiation proceeds before it ends. In the electronic commerce environment, both parties want to make an agreement in a short period of time as far as possible; 2) negotiation total utility, which refers the sum of utility of both parties. Negotiation should have win-win results, therefore, the higher the total utility, the better.

In order to evaluate the proposed tri-training based negotiation methods, we compare our method with the following two negotiation model: 1) KDE based negotiation method [10]; 2) SVM based negotiation method [12].

Experiment 1. In this experiment, we compare total negotiation utility between tri-training, SVM and KDE based negotiation model. Training sample set is extracted from history negotiation data. The trained classifier is used to simulate negotiation process for 100 times. The average total negotiation utility is listed in table 1.

Seen from table 1, with the increasing number of training samples, the average total negotiation utility of all methods increase too. This is because the increasing number of training samples can improve the learning accuracy. The accurate estimation of negotiation preference can improve negotiation process, thus increase

© 2014 ACADEMY PUBLISHER

total negotiation utility. Among them, Tri-training and SVM method can make great total negotiation utility (0.923 and 0.804) in small labeled sample set (50). This is because Tri-training and SVM is a discriminated learning machine, which can maintain better effect in small sample set. With the increase of the number of sample points, the effect of Tri-training and SVM method is not obviously increased, this is because the new sample points are not support vector, and only support vector can improve prediction effect. In small sample set, KDE method has low total negotiation utility (0.583), this is because the KDE is a productive learning machine, relying on the statistical laws of sample points to improve classification precision. Small sample set can't reflect statistical laws, therefore can't guarantee learning accuracy. With the increasing number of sample points, the effect of KDE method is greatly improved, which makes the total negotiation utility increased greatly. Tri-training, as a semi-supervised machine learning method, has more accurate prediction ability than SVM in the same sample set.

TABLE II.
DATA COMPARISON OF TRI-TRAINING, SVM AND KDE BASED NEGOTIATION MODEL

The number of Training samples	Average total negotiation utility		
	Tri-training	SVM	KDE
50	0.923	0.804	0.583
100	0.951	0.834	0.656
150	0.974	0.871	0.719
200	1.011	0.896	0.781
250	1.021	0.917	0.838

Experiment 2. In this experiment, we compare average negotiation round and success ratio between Random, KDE, SVM and tri-training based negotiation model. Random method use random concession strategy.

TABLE I.
DATA COMPARISON OF RANDOM, KDE, SVM AND TRI-TRAINING BASED NEGOTIATION MODEL

Negotiation method	Average negotiation rounds	Success ratio
Random	37.9	55.7%
KDE	21.3	69.2%
SVM	19.5	79.6%
Tri-training	18.6	82.7%

Seen from table 2, Random method has the maximum number of negotiation round; its negotiation success ratio is also low. In the KDE and SVM method, the number of negotiation round significantly reduced; its negotiation success ratio also have improved obviously. When using Tri-training method, two indicators achieve optimal value. This is because the method is the most precise way to estimate negotiation preference, which can rapidly reach an agreement, and at the same time, improve the success ratio of negotiation.

IV. CONCLUSIONS

This paper makes use of the tri-training learner to predict negotiation opponent's negotiation preference. An optimization process is used to produce win-win optimal negotiation counter proposal. Experimental data show that the new method improves the negotiation result. In the next study, we will introduce resemble learning method, which guarantees the accuracy of negotiation preference learning and improves the negotiation total utility in the absence of labeled sample point.

ACKNOWLEDGMENT

This work was supported by grants from Zhejiang Provincial Natural Science Foundation of China (LQ13F020015), Research program supported by the department of education in ZheJiang Province (No.Y201016929)

REFERENCES

[1] N. R. Jennings, P. Faratin, A. R. Lomuscio, “Automated negotiation: prospects, methods and challenges” [J], Group Decision and Negotiation, 10(2), 2001, 199-215

[2] A. Rosenfeld, I. Zuckerman, A. Azaria, “Combining psychological models with machine learning to better predict people’s decisions” [J], Synthese, 189(1), 2012, 81-93

[3] D. P. Twitchell, M. L. Jensen, D. C. Derrick, “Negotiation outcome classification using language features” [J], Group Decision and Negotiation, 22(1), 2013, 135-151

[4] Yanbin Peng, Beishui Liao, Zhijun Zheng, “Negotiation model based on semi-supervised opponent's negotiation preference learning” [J], Computer Integrated Manufacturing Systems, 18(9), 2012, 2082-2090

[5] Yanbin Peng, Jieqing Ai, “Markov chain monte carlo method based bilateral multi-issue negotiation model” [J], Computer Integrated Manufacturing Systems, 17(9), 2011, 2044-2050

[6] Yanbin Peng, Jieqing Ai, “Relevance vector machine based negotiation decision model” [J], Journal of Nanjing University of Science and Technology, 36(4), 2012, 600-605

[7] Yanbin Peng, Zhijun Zheng, Chengbo Yu, “Automated negotiation decision model based on classifier fusion” [J], Journal of Shanghai Jiao Tong University, 47(4), 2013, 644-649

[8] Yanbin Peng, Jieqing Ai, Jiming Li, “Negotiation framework driven by active learning of opponent's preference” [J], Journal of Beijing University of Posts and Telecommunications, 35(5), 2012, 22-25

[9] Yu Cheng, Ji Gao, Huamao Gu, “Automated negotiation decision model based on machine learning” [J], Journal of Software, 20(8), 2009, 2160-2169

[10] R. M. Coehoorn, N. R. Jennings, “Learning on opponent's preferences to make effective multi-issue negotiation trade-offs” , Proceedings of the 6th International Conference on Electronic Commerce, 2004, 59-68

[11] K, Hindriks, D. Tykhonov, “Opponent modelling in automated multi-issue negotiation using bayesian

© 2014 ACADEMY PUBLISHER

learning", Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems, 2008, 331-338
[12] Yu Cheng, Ji Gao, Huamao Gu, "Negotiation decision model based on learning of opponent's attitudes" [J], Journal of Zhejiang University(Engineering Science), 42(10), 2008, 1676-1680
[13] Zhihua Zhou, Ming Li, "Tri-training: Exploiting unlabeled data using three classifiers" [J], IEEE Transactions on Knowledge and Data Engineering, 17(11), 2005, 1529-1541
[14] Tianhao Sun, Qingsheng Zhu, Yunni Xia, Feng Cao, "A Bilateral Price Negotiation Strategy Based on Bayesian Classification and Q-learning" [J], Journal of Information and Computational Science, 8(13), 2011, 2773-2780
[15] Noam Peled, Yaakov Kobi Gal, Sarit Kraus, "An agent design for repeated negotiation and information revelation with people", Proceedings of the 2013 International Conference on Autonomous Agents and Multiagent Systems, 2013, 1303-1304
[16] S.C.Ng, M.N.Sulaiman, M.H.Selamat, "Machine learning approach in optimizing negotiation agents for e-commence" [J], Information Technology Journal, 8(6), 2009, 801-810
[17] Weiguo Zhao, "BP Neural Network based on PSO Algorithm for Temperature Characteristics of Gas Nanosensor" [J], Journal of Computers, 7(9), 2012, 2318-2323
[18] Lian Ye, Yong-kang Xing, Wei-ping Xiang, "An Artificial Immune Classification Algorithm based on Particle Swarm Optimization" [J], Journal of Computers, 8(3), 2013, 772-778
[19] Baojian Zhang, Fugui Chen, Linfeng Bai, "An Improved Partical Swarm Optimization Algrithm" [J], Journal of Computers, 6(11), 2011, 2460-2467

Zhijun Zheng received his Ph.D. degree from Xi'an Jiaotong University. He has been associate professor at Zhejiang University of Science and Technology.

His research interests include machine learning, computer architecture, software engineering, multimedia analysis retrieval, computer animation image retrieval and statistical learning.

Yanbin Peng received his Ph.D. degree from the College of Computer Science and Technology, Zhejiang University. He has been associate professor at Zhejiang University of Science and Technology.

His research interests include artificial intelligence, machine learning, data mining, pattern recognition and image retrieval.

© 2014 ACADEMY PUBLISHER