

An E-Negotiation Agent Using Rule Based and Case Based Approaches: A Comparative Study with Bilateral E-Negotiation with Prediction

Sheetal R. Vij¹, Amruta More¹, Debajyoti Mukhopadhyay², Avinash J. Agrawal³

¹Department of Computer Engineering, Maharashtra Institute of Technology, Pune, India

²Department of Information Technology, Maharashtra Institute of Technology, Pune, India

³Department of Computer Science and Engineering, Ramdeobaba College of Engineering and Management, Nagpur, India

Email: sheetal.sh@gmail.com, moreamruta930@gmail.com, debajyoti.mukhopadhyay@gmail.com, agrawalaj@rknc.edu

Received 20 August 2015; accepted 9 October 2015; published 12 October 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

The research in the area of automated negotiation systems is going on in many universities. This research is mainly focused on making a practically feasible, faster and reliable E-negotiation system. The ongoing work in this area is happening in the laboratories of the universities mainly for training and research purpose. There are number of negotiation systems such as Henry, Kasbaah, Bazaar, Auction Bot, Inspire, and Magnet. Our research is based on making an agent software for E-negotiation which will give faster results and also is secure and flexible. The negotiation partners and contents between the service providers change frequently. The negotiation process can be transformed into rules and cases. Using these features, a new automated negotiation model for agent integrating rule based and case based reasoning can be derived. We propose an E-negotiation system, in which all product information and multiple agent details are stored on the cloud. An E-negotiation agent acts as a negotiator. Agent has user's details and their requirements for a particular product. It will check rules based data whether any rule is matching with the user requirement. An agent will see case based data to check any similar negotiation case matching to the user requirement. If a case matches with user requirement, then agent will start the negotiation process using case based data. If any rule related requirement is found in the rule base data, then agent will start the negotiation process using rule based data. If both rules based data and cases based data are not matching with the user requirement, then agent will start the negotiation process using Bilateral Negotiation model. After completing negotiation process, agent gives feedback to the user about whether negotiation is successful or not. The product details,

How to cite this paper: Vij, S.R., More, A., Mukhopadhyay, D. and Agrawal, A.J. (2015) An E-Negotiation Agent Using Rule Based and Case Based Approaches: A Comparative Study with Bilateral E-Negotiation with Prediction. *Journal of Software Engineering and Applications*, 8, 521-530. <http://dx.doi.org/10.4236/jsea.2015.810049>

rule based data, and case based data will be stored on the cloud. So that system automatically becomes flexible. We also compare E-negotiation agent automated negotiation and behavior prediction system to prove that using rule based and case based approaches system should become fast.

Keywords

Automated Negotiation, Multi-Agent, Rule Based Reasoning, Case Based Reasoning, Cloud Computing

1. Introduction

Practically “negotiation” can be defined as an iterative process which aims to achieve a mutually beneficial deal for the seller and buyer [1]. The automation saves human negotiation time and computational negotiators are better at finding deals in combinatorial and strategically complex settings. The rapid success of online auctions clearly shows that E-negotiation will eventually become the basis of e-commerce. Whether it is a case of B to B purchase or a case of online shopping, it is required to make the traditional negotiation pricing mechanism automated and intelligent [2].

Cloud computing is innovation that uses advanced computational power and improved storage capabilities. Cloud computing is a new processing scheme in which computer processing is performed in the network. This means that users need not concern themselves with the processing details. Although Cloud computing enables more flexible, easier and faster computing [3].

Case Based Reasoning (CBR) is a problem solving paradigm where the solution of new problem is based on solution of similar past problem. We use Rule Based Reasoning (RBR) concept, where there are some rules such as discount, festival offers etc.

In this system, we are introducing an E-negotiation agent based system using Rule Based Reasoning and Case Based Reasoning. Due to the use of Rule Based Reasoning and Case Based Reasoning, system becomes faster. In this system, all data that is product detail, case base data and rule base data are stored on cloud. Therefore system becomes flexible.

2. Literature Review

In this section, we are presenting literature survey related to only rule based and case based reasoning. In our previous paper [4], we have referred ten papers related to automated negotiation in the literature survey.

Liu Xiaowen and Yu Jin [5] introduced automated negotiation model for tourism industry. To improve the negotiation efficiency and success rate, this system proposed RBR and CBR. The model employs CBR method to support an automated negotiation by past successful negotiation cases used for those negotiation partners that have no contract rule existing in each other. This system does not support multi party multi issue negotiation.

Mohammad Irfan Bala, Sheetal Vij and Debajyoti Mukhopadhyay [6] introduced, E-negotiation system with behavior prediction. This work reviews the various methods used for predicting the opponent’s behavior and then proposes architecture for behavior prediction using artificial neural networks. It proposes the use of database for storing the results and suggests various issues that can be taken into consideration while predicting the opponent’s behavior.

Mira Vrbaski and Dorina Petriu [7] proposed Context-aware systems which use Rule Based Reasoning engines for decision making without involving explicit interaction with the user. It is difficult to rank suitable solutions based on unclear, qualitative criteria with a rule based approach, while rule based systems excel in filtering out unsuitable solutions based on clear criteria.

Leen-Kiat Soh and Costas Tsatsoulis [8] use Case Based Reasoning (CBR) and utility to learn, select, and apply negotiation strategies. Agent uses Case Based Reasoning (CBR) approach to solve new problem of negotiation strategy which based on previous similar past problem. Agent also learns from its previous negotiation experience.

P. Maes, R. Guttman, A. Moukas [9] introduced a Kasbah negotiation model. In this system, agents can only

negotiate over the single issue of price. However, B2B negotiations often involve multiple issues. Moreover, the Kasbah agents can only act according to one of their pre-defined negotiation strategies which may not lead to the optimal negotiation results.

P. Wurman, M. Wellman, W. Walsh [10] introduced the Michigan AuctionBot is a general purposed Internet-based auction server hosted by the University of Michigan. Sellers can create new auctions on AuctionBot by choosing from a set of pre-defined auction types and then enter their specific auction parameters such as clearing time, minimum bid increment and whether proxy bids are allowed. E-bay is the example of AuctionBot negotiation system.

Some of above papers support multi party multi issue negotiation Rule Based Reasoning and Case Based Reasoning. Our negotiation system is a bilateral, muti-party, multi issue negotiation model. In this system, buyer and seller negotiate o multiple issues at a time and when both buyer and seller comes to final decision, then only negotiation process will be stopped.

3. Proposed System

3.1. Problem Statement

For negotiation process, more than two parties come together to reach mutually beneficial outcome. For similar negotiation case, there is no need to do negotiation process again due to use of case based approach. We also provide some rules related to negotiation process which makes successful negotiation process. Due to use of rule based and case based approaches, negotiation system become faster. We also use Cloud computing concept, to store all organizations' product data. Hence, maintenance of organizations' data is reduced.

3.2. Mathematical Model

The given mathematical model is for bilateral negotiations where an agent can negotiate about multiple issues. It also supports learning from the previous negotiation rounds. The mathematical model for the proposed work is as follows:

$$N = \{A, U, R, D, T\}$$

A: *A* is an agents which will participate in the negotiation.

U: The set of users that will participate in the negotiation. This set consists of both sellers and buyers.

$U = \{u_1, u_2, \dots, u_m\}$ represents *m* number of users.

R: The set of requirement which are given by users to their respective agents. This set consists of both sides' seller side and buyer side requirements.

$R = \{r_1, r_2, \dots, r_i\}$ represents *i* number of requirements that is “*R*” represents total number of requirements which are available at current stage.

D: The set of database which is used for matching the users' requirement with rule based and case based databases.

T: The time limit for the negation.

Whole process of negotiation should complete before the time limit.

Constrains:

In previous we can use *R* for set of requirements but requirement of particular user is become a constrain.

Algorithms:

A_1 = A single user gives his requirements to respective agent.

$$N = \{A, U, R\}$$

$$A_1 = \{A, U_1, R_1\}$$

Here, mapping of R_1 to *R*, U_1 to *U*.

A_2 = Agent checks the user requirements with rule based and case based databases

A_3 = After checking databases, agent will start actual negotiation process.

A_4 = Total time of negotiation process should be less than or equal to *T*.

$$T \leq N_{\text{time}}$$

where, N_{time} is the total time of negotiation process.

Flow diagram of mathematical model is shown in **Figure 1**.

4. System Architecture

The architecture of E-negotiation agent system is shown in **Figure 2**.

Working of E-negotiation agent system:

1) Databases on cloud

- Product detail: used to store organizations' product detail like product name, price, quantity;
- Case base: used to store experience of agent about negotiation process;
- Rule base: used to store rules related to product like discount, festival offers.

2) Seller and Buyer: Users of the system

3) Agent: An agent acts as a negotiator

Negotiation Process: After receiving Seller's and Buyer's requirement, agent will start negotiation process.

For negotiation process, agent will check three conditions.

- Case based: An agent will see case based data to check any similar negotiation case is matching to the user requirement. If a case matches with user requirement, then agent start the negotiation process using case based data.
- Rule based: It will check rules based data whether any rule is matching with the user requirement. If any rule related requirement is found in the rule base data, then agent will start the negotiation process using rule based data.
- Bilateral Negotiation model: If both rules based data and case based data are not matching with the user requirement, then agent will start the negotiation process using Bilateral Negotiation model.

4) Feedback: After completing negotiation process, agent gives feedback to the user about whether negotiation is successful or not.

Flow chart of E-negotiation agent system is shown in **Figure 3**.

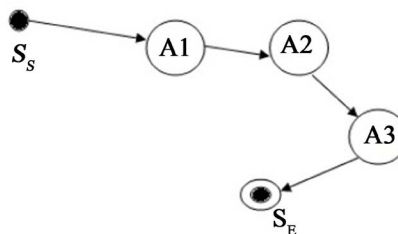


Figure 1. Flow diagram of mathematical model of E-negotiation agent system.

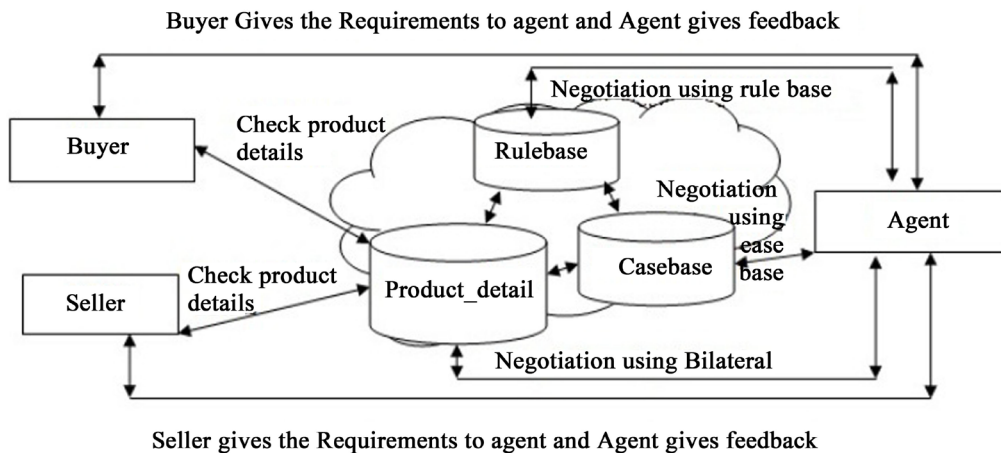


Figure 2. System architecture of E-negotiation agent system.

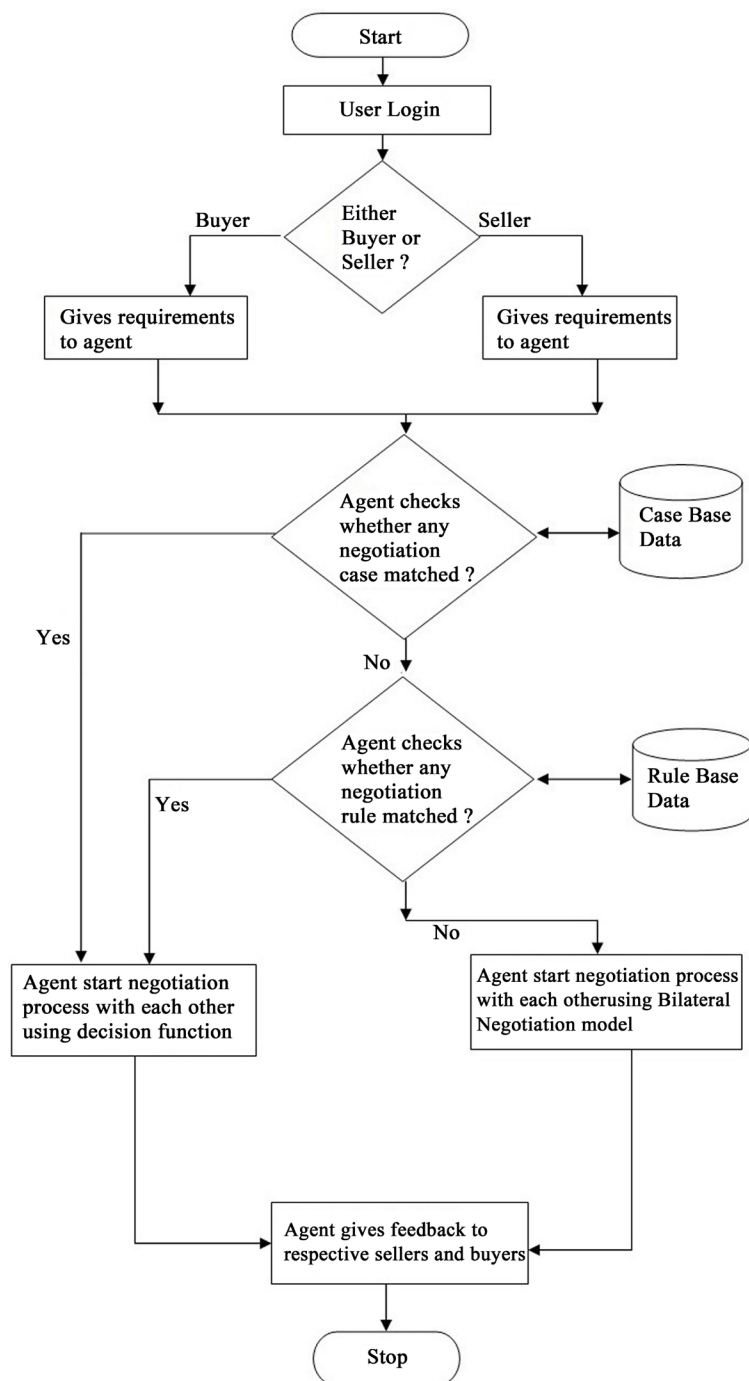


Figure 3. Flow chat of E-negotiation agent system.

Components of the System

For this system, we can use three components. Using these components, system becomes easy to use and work efficiently.

1) *Store data on cloud*: For storing data, we can use Open Stack cloud. Open Stack is a Cloud computing project to provide an IaaS. It is free open source software released under the terms of the Apache License.

In this module, all data which is required for negotiation process like product data, rule base data and case base data is stored on cloud.

2) *Negotiation process*: For negotiation process [4], seller and buyer give their requirements to agent in encrypted format that is generate the hash code of that requirement and encrypt that hash code using user's public key. User's public key is known to Agents.

$$\text{Buyer or seller requirement} = E\{H(m), U_{pk}\}$$

where, for generating hash function MD5 algorithm is used.

U_{pk} is user's public key.

We can use encryption for security purpose, in this process, we can use digital signature concept.

Same as seller, buyer can do same process for generating hash code of his requirement.

After getting requirement, agent decrypt the hash code using his private key and calculate the hash code for checking whether this message comes from appropriate seller or buyer and whether it is modified or not.

$$\text{Agent receive requirements} = D\{A_{pri}, H(m)\}$$

where, A_{pri} is agent's private key.

After getting the requirement, agent checks the user requirements whether any rule related to requirement is found in the rule base data then only agent will start the negotiation process using decision function [11].

- **Decision Function**: After getting the requirement, agent checks the user requirements whether any rule related to requirement is found in the rule base data then only agent will start the negotiation process using decision function. The negotiation .model introduces a decision function [11] for supplier and buyer first. Seller's agent responds at t_n to buyer's agent offer sent at time t_{n-1} and total profit is defined as Equation (1). Similarly, buyer's agent also has a decision function to respond at t_n to seller's agent offer sent at time t_{n-1} and total cost is shown in Equation (2).

$$D^S(t_n) = \begin{cases} \text{terminate} & \text{if } t_n > t_{\max} \\ \text{accept} & \text{if } TP(x_J^{B \rightarrow S}(t_{n-1})) \leq TP(x_J^{S \rightarrow B}(t_n)) \\ \text{otherwise} & (x_J^{S \rightarrow B}(t_n)) \end{cases} \quad (1)$$

$$D^B(t_n) = \begin{cases} \text{terminate} & \text{if } t_n > t_{\max} \\ \text{accept} & \text{if } TC(x_J^{S \rightarrow B}(t_{n-1})) \geq TP(x_J^{B \rightarrow S}(t_n)) \\ \text{otherwise} & (x_J^{B \rightarrow S}(t_n)) \end{cases} \quad (2)$$

The buyer uses Equation (2) to decide whether he should accept suppliers' proposal or not. Similarly, using a decision function Equation (1) a supplier can make decision.

- **Bilateral Negotiation model**: If both rules based data and case based data are not matching with the user requirement, then Buyer's agent and Seller's agent will start the negotiation process with each other using Bilateral Negotiation model.

For negotiation process, we can use the Bilateral Negotiation model [12].

Let x represents the buyer agent ($x \in \{x_1, x_2, \dots, x_m\}$) and y ($y \in \{y_1, y_2, \dots, y_m\}$) be the supplier agent. And let then i ($i \in \{i_1, i_2, \dots, i_n\}$) be the issues under negotiation, such as price, volume, duration, quality and so on. Each agent assigns to each issue i a weight W_i , denoting the relative importance of that issue to the agent. Here, we consider quality as a weight. Hence, W_i^x represents the importance of issue i to agent x , therefore the overall utility function of an offer O is

$$U(O) = \frac{\sum_{i=1}^m W_i u_i(x_i)}{\sum_{i=1}^m W_i} \quad (3)$$

where the overall utility is denoted as $U(O)$ for the offer $O = [O_1, \dots, O_m]^T$, and $u_i(x_i)$ is the individual utility function for issue i for $u_i \in [0, 1]$ and the preference degree of an agent to an issue I is denoted as $W_i \in [0, 9]$.

Each agent also specifies a minimum acceptable utility level [U_{\max} , U_{\min}] to determine if an offer is acceptable. Hence, for benefit-oriented, the utility function $U_i(x_i)$ is computed as follows :

$$U_i(x_i) = \frac{x_i - x_{\{\min\}}}{x_{\{\max\}}^i - x_{\{\min\}}^i} \quad (4)$$

For cost oriented however, the utility function can be as follows:

$$U_i(x_i) = 1 - \frac{x_i - x_{\{\min\}}}{x_{\{\max\}}^i - x_{\{\min\}}^i} \quad (5)$$

3) *Feedback*: When negotiation process was finished. Agent gives feedback to appropriate seller or buyer about negotiation whether it is successful or not.

5. Experimental Analysis and Results

We implemented E-negotiation agent system using rule based and case based approaches. We took a system (automated negotiation and behavior prediction) [6] as an existing system.

Mohammad Irfan Bala implemented an automated negotiation and behavior prediction system. In this system, during negotiation the agent will store the offers received and predict the preferences of the opponent based on these offers. The negotiation protocol determines the overall order of actions during a negotiation and the agents are obliged to stick to this protocol. In the bilateral alternating offers protocol two parties—agent A and agent B—take turns. Agent A starts the negotiation. Each turn an agent presents one of the three possible actions:

Accept: This action indicates that agent accepts the opponent's last bid.

Offer: This action represents the bid made by an agent.

End negotiation: This action indicates that the agent terminates the negotiation.

We compared E-negotiation agent using rule based and case based approaches system with automated negotiation and Behavior Prediction system based on response time.

Table 1 shows the response time values of both systems. We calculated response time value of both the systems up to 300 users. Therefore we can say that, using rule based and case based approaches E-negotiation process requires less time as compared to existing system.

Figure 4 shows the comparison graph based on response time.

In E-negotiation agent using rule based and case based approaches system, all the data which is required for E-negotiation process such as product detail, rule base data, case base data is stored on cloud. Automated negotiation and Behavior Prediction system does not use Cloud computing concept to store negotiation process.

We calculated CPU utilization (in percentage) and Memory utilization (in percentage) of both systems up to 300 users.

Table 2 shows the Memory utilization values (in percentage) of both systems.

Therefore we can say that, due to use of Cloud computing E-negotiation agent using rule based and case based approaches system requires less CPU utilization and Memory utilization as compared to existing system.

Table 3 shows the CPU utilization values (in percentage) of both systems.

Figure 5 shows the comparison graph based on Memory utilization.

Figure 6 shows the comparison graph based on CPU utilization.

Table 1. Response time values of both systems.

No. of Users	Response Time (E-Negotiation Agent)	Response Time (Existing System)
50	256	360
100	591	641
150	723	803
200	1138	1157
250	1364	1513
300	1681	1741

Table 2. Memory utilization values of both systems.

No. of Users	Memory% (E-Negotiation Agent)	Memory% (Existing System)
50	9%	12%
100	10%	13%
150	12%	16%
200	15%	19%
250	18%	23%
300	21%	27%

Table 3. CPU utilization values of both systems.

No. of Users	CPU% (E-Negotiation Agent)	CPU% (Existing System)
50	17%	20%
100	34%	43%
150	52%	61%
200	66%	73%
250	71%	82%
300	89%	93%

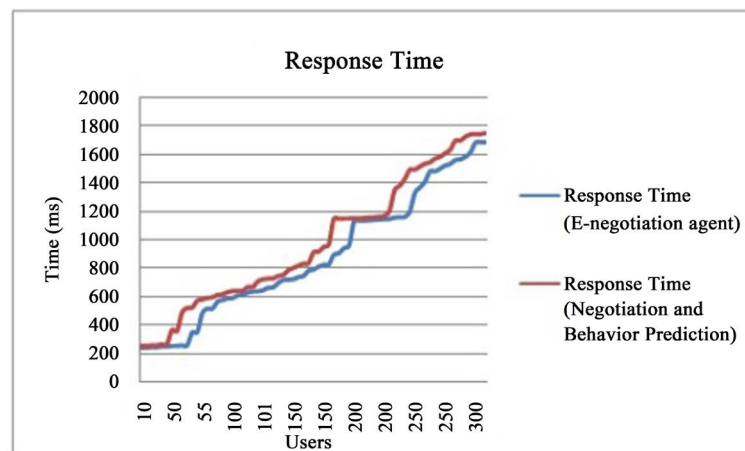


Figure 4. Comparison graph based on response time.

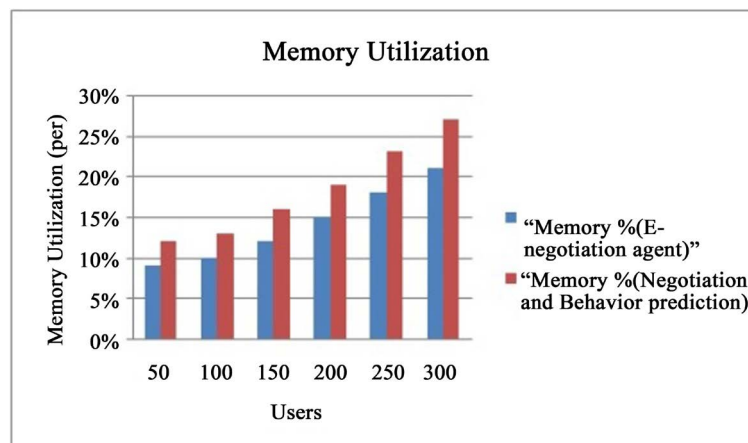


Figure 5. Comparison graph based on memory utilization.

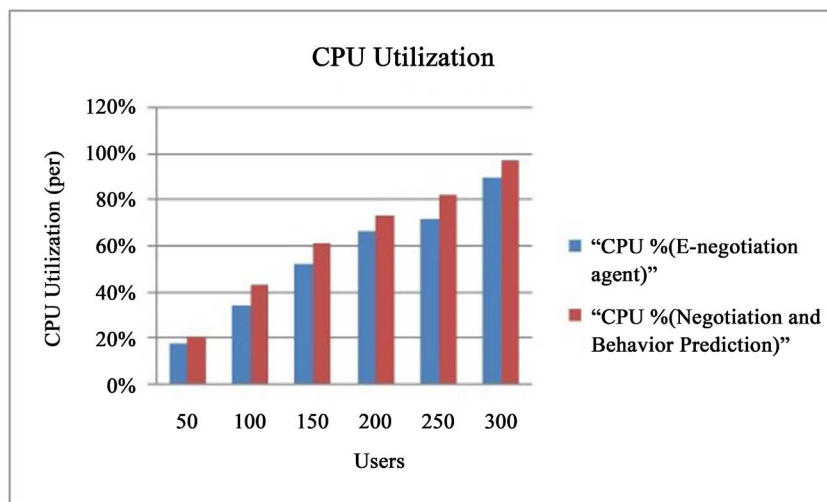


Figure 6. Comparison graph based on CPU utilization.

6. Conclusion and Future Scope

In order to make faster E-negotiation process, we use the rule base and case based approaches. Two or more parties are coming together during the negotiation process. And there are organizations to maintain data of the negotiation process and product data. But this maintenance is a very tedious job. In order to overcome this problem, all organizations' product data are stored on cloud. So that system should become flexible.

This system is a Bilateral Negotiation model. In future this system can be implemented as multilateral negotiation model, behavior prediction and also uses the concept of expert system for increasing success rate of negotiation process.

References

- [1] Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra, C. and Wooldridge, M. (2002) Automated Negotiation: Prospects, Methods and Challenges. *Group Decision and Negotiation*, **10**, 199-215. <http://dx.doi.org/10.1023/A:1008746126376>
- [2] Mukun, C. (2010) Multi-Agent Automated Negotiation as a Service. *7th International Conference on Service System and Service Management (ICSSSM)*, Tokyo, 28-30 June 2012, 308-313.
- [3] Singh, J., Kumar, B. and Khatn, A. (2012) Securing Storage Data in Cloud Using RC5 Algorithm. *International Journal of Advance Computer Research*, **2**, 94-98.
- [4] More, A., Vij, S. and Mukhopadhyay, D. (2013) Agent Based Negotiation Using Cloud—An Approach in E-Commerce. *Proceedings of 48th Annual Convention of the Computer Society of India*, Visakhapatnam, 13-15 December 2013, 489-496.
- [5] Liu, X.W. and Yu, J. (2012) Hybrid Approach Using RBR and CBR to Design an Automated Negotiation Model for Tourism Companies. *2012 International Conference on Management of E-Commerce and E-Government*, **21**, 197-201.
- [6] Bala, M.I., Vij, S. and Mukhopadhyay, D. (2013) Negotiation Life Cycle: An Approach in E-Negotiation with Prediction. *Proceedings of 48th Annual Convention of the Computer Society of India*, Visakhapatnam, 13-15 December 2013, 505-512.
- [7] Vrbaski, M. and Petriu, D. (2012) Tool Support for Combined Rule-Based and Goal-Based Reasoning in Context-Aware Systems. *Requirement Engineering Conference 2012*, Chicago, 24-28 September 2012, 335-336.
- [8] Soh, L.-K. and Tsatsoulis, C. (2001) Agent-Based Argumentative Negotiations with Case-Based Reasoning. AAAI Technical Report FS-01-03.
- [9] Maes, P., Guttman, R. and Moukas, A. (1999) Agents That Buy and Sell. *Communications of the ACM*, **42**, 81-91. <http://dx.doi.org/10.1145/295685.295716>
- [10] Wurman, P., Wellman, M. and Walsh, W. (1998) The Michigan Internet AuctionBot: A Configurable Auction Server for Human and Software Agents. In: Sycara, K.P. and Wooldridge, M., Eds., *Proceedings of the 2nd International*

Conference on Autonomous Agents, ACM Press, New York, 301-308. <http://dx.doi.org/10.1145/280765.280847>

- [11] Rau, H., Chen, C.-W. and Shiang, W.-J. (2009) Development of an Agent-Based Negotiation Model for Buyer-Supplier Relationship with Multiple Deliveries. *Proceedings of the 2009 IEEE International Conference on Networking, Sensing and Control*, Okayama, 26-29 March 2009, 308-312.
- [12] Ateib, M.T. (2010) Agent Based Negotiation in E-Commerce. *International Symposium on Information Technology* 2010, **2**, 861-868.