30-8-2004

A Unified Logical Model for CBR-based E-commerce Systems

Zhaohao Sun
University of Wollongong, zsun@uow.edu.au

G. Finnie
Bond University

A Unified Logical Model for CBR-based E-commerce Systems

Abstract

This paper will examine new issues resulting from applying CBR in e-commerce and propose a unified logical model for CBR-based e-commerce systems (CECS) which consists of three cycles and covers almost all activities of applying CBR in e-commerce. This paper also decomposes case adaptation into problem adaptation and solution adaptation, which not only improves the understanding of case adaptation in the traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce. It then investigates CBR-based product negotiation. This paper thus gives insight into how to use CBR in e-commerce and how to improve the understanding of CBR with its applications in e-commerce from a logical viewpoint.

Keywords

Case based reasoning, e-commerce, intelligent support, CBR-based e-commerce systems

Disciplines

Business | Social and Behavioral Sciences

Publication Details

This article was originally published as: Sun, Z & Finnie, G, A Unified Logical Model for CBR-based E-commerce Systems, International Journal of Intelligent Systems, 2005, 20(1), 29-46. Copyright 2005 John Wiley & Sons.

A Unified Logical Model for CBR-based E-commerce Systems

Zhaohao Sun†, Gavin Finnie*

School of Economics and Information Systems, University of Wollongong,
Wollongong NSW 2522, Australia
Email: zsun@uow.edu.au

† Author to whom correspondence should be addressed; zsun@staff.bond.edu.au

* Faculty of Information Technology, Bond University
Gold Coast Qld 4229, Australia
gfinnie@staff.bond.edu.au

Abstract. This paper will examine new issues resulting from applying CBR in e-commerce and propose a unified logical model for CBR-based e-commerce systems (CECS) which consists of three cycles and covers almost all activities of applying CBR in e-commerce. This paper also decomposes case adaptation into problem adaptation and solution adaptation, which not only improves the understanding of case adaptation in the traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce. It then investigates CBR-based product negotiation. This paper thus gives insight into how to use CBR in e-commerce and how to improve the understanding of CBR with its applications in e-commerce from a logical viewpoint.

1. Introduction

Adding intelligence to e-commerce systems as well as other Internet applications is an important role for CBR in e-commerce.[22][23][24][25] E-commerce sets out to sell products without the intervention of a sales assistant and in the absence of human sales assistants.[13] Instead, there is a need for e-commerce to have intelligent software assistants to lubricate its sales process. CBR has proved an available technology to create these sales assistants,[28][29] since catalogue data and data on user behavior and preferences are basically available from traditional commerce. For example, one can use cases to describe commodities on sale, and use CBR to identify the case configuration that meets the customers' requirements.

CBR has found many applications in e-commerce as an assistant in e-commerce stores and as a reasoning agent for online technical support,[13] as well as an intelligent assistant for sales support or for e-commerce travel agents.[2][18] The strength of CBR in this area not only stems from its reuse of the case base, or experience base and case adaptation associated with a particular application but also from that CBR provides a comprehensive model for almost all e-commerce applications, which corresponds to a unified logical model (see later).

More recently, product search, product selection[2] and customization,[21] product recommendation,[6] product configuration, and product negotiation have been drawing increasing attention in research and commercial activities of applying CBR in e-commerce.[4][16][17][19][25] However, it seems that the applications and activities are very discrete or isolated from a CBR perspective. The relationship between these applications and the traditional CBR-cycle[1][6][9][27] is also not clear. In particular, the logical correspondence relationship between the CBR-cycle and e-commerce applications based on CBR-techniques has not been drawn any attention in CBR related research studies and publications, although CBR is, first of all, a reasoning paradigm. For example, Refs. [2][3][6][28] have no attempt to build the logical correspondence between the CBR cycle and e-commerce applications, although they try to model some aspects in e-commerce such as product selection, retrieval and recommendation from a CBR viewpoint. Furthermore, there do not exist satisfactory answers to the following two important questions:

- Why can CBR be applied for e-commerce?
- Where can CBR really be applied in e-commerce applications?

Providing examples of applying CBR in e-commerce can only be considered as a partial answer to the above questions. Therefore, the first goal of this paper is to propose a unified logical model for a CBR-based e-commerce system (CECS) which covers almost all activities of applying CBR in e-commerce and give new insight into the traditional CBR cycle.[1][9][11] From this unified logical model we can establish why CBR can be applied in e-commerce, and where CBR can really be applied in e-commerce applications. This paper then investigates CBR in product negotiation. This paper thus gives insight into how to use CBR in e-commerce and how to improve the understanding of CBR with its applications in e-commerce. These two issues are complementary. Almost all researchers in this area (e.g. Refs. [6][15][29]) have focused on the first part, while the second part has been largely neglected. Another of the contributions of this paper is the decomposition of case adaptation into problem adaptation and solution adaptation, which not only improves the understanding of case adaptation in traditional CBR, but also facilitates the refinement of activity of CBR in e-commerce and intelligent support for e-commerce. Applying CBR in e-commerce will be treated in a unified way at three levels: logical level, CBR-level, and e-commerce level.

The rest of this paper is organized as follows: Section 2 examines case-based reasoning as a process reasoning. Section 3 discusses a general model of web-based CBR systems. Section 4 proposes a unified logical model of CBR-based e-commerce systems (CECS), in which we also investigate problem adaptation and solution adaptation. Section 5 examines case-based negotiation. Section 6 ends this paper with some concluding remarks.

It should be noted that e-commerce is a very broad area of research encompassing e-marketplaces, customer relationship management, vendor management, logistics, supply chain management, agent chain management, distributed workflow and other topics. No single paper can cover all the mentioned areas, although its title includes e-commerce. Neither does this paper. Even so, this paper limits itself to product search, product selection, product retrieval, product recommendation, product configuration (adaptation), and product negotiation, because each of the mentioned aspects corresponds to a certain stage of the CBR cycle.

2. Case-Based Reasoning as Process Reasoning

In this section, we emphasize that case-based reasoning (CBR) is first a kind of reasoning, then it is a process reasoning. The first point has been neglected to some extent in the CBR community. The second point has been proposed by the author.[11] The key idea behind it is that the reasoning-based systems in Artificial Intelligence (AI) are always involved in process reasoning.

CBR is, first of all, a kind of reasoning from its name. As we know, reasoning is an important problem solving tool in mathematics and logic. Reasoning also is an important tool in AI, in which it is based mainly on the reasoning in mathematical logic such as propositional logic and predicate logic.[11] Based on this idea, CBR has a close relation with mathematical logic, which is the fundamental source for any study of applying fuzzy logic to CBR.[11] Therefore, it is necessary to examine CBR from a logical viewpoint, which has been basically neglected in research and development of CBR.[11][25] For example, one cannot find any logical model in some CBR related books.

Furthermore, CBR is a reasoning paradigm based on previous experiences or cases; that is, a CBR system solves new problems by adapting solutions that were used to solve old problems.[27] Therefore, we call CBR the form of experience-based reasoning at a general level, briefly,[23]

$$\text{CBR} := \text{Experience-based reasoning} \tag{1}$$

In business activities, it is usually true that "Two cars with similar quality features have similar prices." This is a kind of experience-based reasoning which is, essentially, a kind of similarity-based reasoning. In other words, similarity-based reasoning can be considered as a special and operational form of experience-based reasoning. Therefore, CBR can be considered as a kind of similarity-based reasoning from a logical viewpoint.[11]

$$\text{CBR} : = \text{Similarity-based reasoning} \tag{2}$$

Similar to the inference engine (IE) and knowledge base in expert systems (ES), one can also use CBR engine (CBRE) and a case base (CB) to denote a CBR system (CBRS); that is,

$$\text{CBRS} = \text{CB} + \text{CBRE} \tag{3}$$

where the CBR engine performs similarity-based reasoning, while the inference engine in ESs performs traditional deductive reasoning. One of the most important inference rules for deductive reasoning is *modus ponens*:[11]

$$\begin{array}{l} P \rightarrow Q \\ \underline{P \qquad\quad} \\ \quad \therefore Q \end{array} \tag{4}$$

The inference rule for similarity-based reasoning can be considered as *generalized modus ponens:*[11]

$$\begin{array}{l} P \to Q \\ \underline{P'\quad\quad} \\ \quad \therefore Q' \end{array} \tag{5}$$

Therefore the CBRE is based on Equation (5). In other words, the CBRE is a mechanism for performing similarity-based reasoning.

However, CBR is not only a simple similarity-based reasoning, but it is a kind of process reasoning,[11] which is a more complex reasoning paradigm that is usually used in intelligent systems.

A *process reasoning* is reasoning that infers information about a domain using process or multistage methods and there exists a traditional reasoning paradigm which plays a vital role in every main stage of the process.[11] In what follows of this section, we examine CBR as a process reasoning, which differs from not only traditional mathematical reasoning but also from fuzzy reasoning or similarity-based reasoning.

A typical reasoning in CBR, also named as the CBR cycle,[1][9] mainly consists of (case) Repartition, Retrieve, Reuse, Revise and Retain. Each of these five components is a complex process. For example, case retrieval is a complex operation in the case base. Furthermore, case retrieval and case adaptation are two main stages in CBR, in which similarity-based reasoning

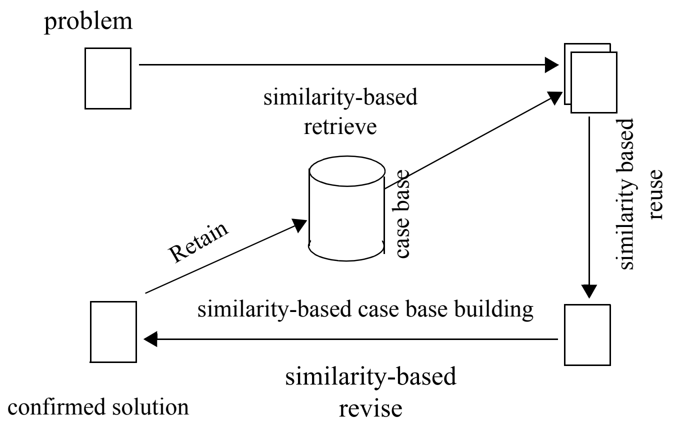

Fig. 1. Overall inference of CBR as a process reasoning after [11]

plays an important role. For instance, case retrieval is based on similarity based reasoning,[7] case adaptation is also based on it, but maybe on a different similarity based reasoning (see later).[11] In fact, case base repartition and building is also based on similarity based reasoning.[8] Thus, CBR is a process reasoning, in which similarity based reasoning dominates each of main stages; that is, case base building, case retrieval, and case adaptation, as shown in Fig. 1.

3. A General Model Of Web-Based CBR Systems

CBR has drawn increasing attention in developing Web-based intelligent systems.[9] Several Web-based systems that use CBR are already in existence.[14] For example, WEBSELL is an intelligent sales assistant for the WWW, which uses case-based retrieval and case-based recommendation.[5][6] A characteristic of these applications is that they involve implementations of existing CBR technology in a Web context: the client has a remote dialogue through the browser with the CBR application at the server side.[29] Based on this idea, this section reviews a general model for Web-based CBR systems after,[14] as shown in Fig. 2. In this model, each client has its own case base (i.e. Client CB) and a browser based interface at the front-end that connects to the server at the back-end; all the case base processing is performed at the back-end. In the distributed architecture the CBR engine is downloaded to the client side to allow for the later stages of processing to be performed there.

The details of the operation of this Web-based CBR system can be explained in the context of electronic sales.[14][25] The interface allows a customer to describe his/her demands p'. This is normalized into a partial case description (which really is a problem description) that is passed to the CBR front-end as a Query context (for brevity, it is still p'). Initially, this will be passed to the CBR back-end to find matching cases (for example, $c = (p, s)$). If too many potential matches are found the CBR engine will identify which feature of the problem descriptions in the matched cases is the most discriminating. This is then passed to the user interface as a Refining Question, which is still denoted as p'. This process is a typical demand adaptation (or problem adaptation, see later). The response to this request for extra information, which is the result of carrying the demand adaptation, is passed to the back-end as a refined Query context, which is still p', for brevity. This process is continued until such time as the Query context is sufficiently discriminating. At this point, matching cases are passed to the user interface as a product recommendation.

In this process, as the Query context is refined, the set of potentially matching cases reduces, for example, $c_1, c_2, \ldots, c_n$ are reduced to $c_1, c_2, \ldots, c_m$, where $m < n$. The advantage

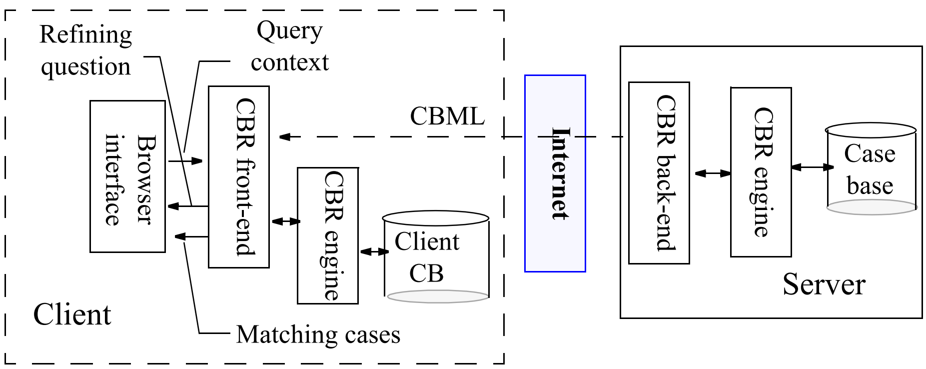

Fig. 2. General architecture of Web-based CBR systems after [14]

of the proposed model is that once this set is sufficiently small it can be passed to the front-end where processing can be completed without further interaction across the network. The decision as to when precisely to do this depends on the size of the cases and the response time across the network.

However, from the viewpoint of CBR, this model has not realized the main CBR-cycle in an explicit way,[9] because it only stressed case retrieval and problem (demand) adaptation. In the following section, this model, in particular the CBR engine, will be improved in order to express how the CBR-cycle corresponds to the activities of the business world including the e-commerce world.

Furthermore, since the end of last century, the relationship between the CBR cycle and e-commerce applications based on CBR has been addressed by several research projects and some publications such as WEBSELL.[2][5][21][28] However, their work are still at an empirical level; that is, they try to provide examples of using CBR terminology to explain the real world scenarios in e-commerce applications. There have been no attempts to provide a unified model for such a relationship from a logical viewpoint, although a model is better than a thousand examples in some cases. In what follows, we will then propose such a model, with which we build a correspondence between the traditional CBR cycle and the logical model of the CBR cycle and their correspondence to the application areas of CBR in e-commerce.

4. A Unified Logical Model of CBR-Based E-Commerce Systems

The above-mentioned model will be examined in the context of electronic sales in a revised way, which constitutes the basis for applying CBR in e-commerce or a CBR e-commerce system (for short, CECS).[25] The aforementioned Web-based CBR system can be considered as a concrete realization of a CECS. The customer describes his demand p' to the CECS through its interface. This demand is normalized into an officially structured problem description p'. Then the CECS uses its similarity metric mechanism[8] (the inference engine based on $\sim$) to search and retrieve its case base, which consists of cases, each of which is denoted $c = (p, q)$, where p is the structured problem description and q is the solution description. In the CECS, *problem description* and *solution description* correspond to *(customer) demand description* and *product description* respectively. The similarity metric mechanism performs similarity-based reasoning, which can be formalized as:[10][11]

$$\frac{P', P' \sim P, P \rightarrow Q, Q \sim Q'}{\therefore\ Q'} \tag{6}$$

where P, P', Q, and Q' represent compound propositions, $P' \sim P$ means that P and P' are similar. Q and Q' are also similar. This is the essence of any similarity-based reasoning.[23] This is also a logical model of both the CBR cycle and CECS.

4.1 A Logical Model for Case Retrieval

In this section, we propose a logical model for case retrieval, which can facilitate the research and development of both CBR and intelligent information retrieval.

From a logical viewpoint, the case search and retrieval process is basically to find the following case set from the case base in the CECS[a],

$$C(p') = \{c \mid c = (p, q), p \sim p'\} = \{c_1, c_2, \ldots, c_n\} \tag{7}$$

where n is a positive integer, c_i, $i = 1, 2, \ldots, n$ are all cases with their problem description

a. Therefore, case selection is one part of case retrieval.[1]

p similar to the current problem description p'. Usually, $C(p') = \{c_1, c_2, \ldots, c_n\}$ satisfies the following property: For any integer i, $1 \le i < n$ and $c_i = (p_i, q_i)$,

$$s((p_i, p') \ge s(p_{i+1}, p')) \tag{8}$$

where $s(\cdot)$ is a similarity metric,[8] which measures the similarity between one object and another.

It should be noted that although there are many research studies that discuss case search and retrieval,[2][5][6][21][28] which correspond to product search and retrieval in CECS, they usually discuss case retrieval in a confusing way, because they essentially compare the problem (which corresponds to the demand description of a customer) with the cases in the case base of the CBR system, based on a similarity-based inference mechanism. Model (7) is the first formulation that separates the case into the problem and the solution in an obvious way.

4.2 A Logical Analysis for Case Reuse and Case Recommendation

If n is reasonably small, then the CECS will directly recommend the product descriptions of $c_1, c_2, \ldots, c_n$, $q_1, q_2, \ldots, q_n$, through the user interface. This process is usually *case reuse,* which is one process stage of the CBR cycle.[11] In practice, this process can be considered as *product search* and *product selection*.[2][21] If n is very large, the CECS has to recommend the product descriptions of the first m cases in $c_1, c_2, \ldots, c_n$; that is, $q_1, q_2, \ldots, q_m$, to the customer, in order to meet the needs of the customer, where $1 \le m < n$. This process can be called *product recommendation*. More generally, although it is still case reuse in the traditional CBR cycle, this process can be considered as *case recommendation*, because the CECS usually provides the customer with not only the recommended products but also the customer demand description. Therefore, case recommendation is more significant than case reuse from a customer viewpoint, while case reuse is more useful from a system developer's viewpoint. Based on this idea, case reuse in the traditional CBR cycle[1] can be considered as a system concept for intelligent system development, to some extent.

Case recommendation has not been mentioned in the CBR-cycle,[1][9] and thus it is a new concept for CBR motivated from CECSs. More specifically, case recommendation is a more specialized form of case reuse. In practice, case recommendation is a more complex process than case reuse. Based on the above discussion, product (case) recommendation is a process after case retrieval. Therefore, case recommendation is a process necessary for applying CBR in e-commerce. In this way, the traditional CBR cycle[1] has been improved by adding case recommendation to it.

It should be noted that product recommendation has been described in many CBR application publications.[2][5][6][21] However, they have not given a clear explanation of the relationship between case reuse and case recommendation, and in particular, they have not treated them from a logical viewpoint, which will be addressed in next section.

4.3 A Cyclical Model for Case Recommendation

In this section, we will further examine case recommendation and propose a cyclical model for case recommendation.

After the customer obtains the recommended product descriptions from the CECS, he will evaluate them and then select one of the following:

1. Accept one of the recommended products, q_k, and order it, where $1 \le k \le m$
2. Adjust his demand descriptions p' and then send them to the CECS

3. Refuse the recommended products and leave the CECS.

It is obvious that among these three cases only the first two require further discussion:

For the first case, the deal was successfully done and the CECS routinely updates[a] the successful case $c_k = (p_k, q_k)$ in the case base. At the same time, the CECS has reused the case successfully. This means that the CECS has performed *case reuse* and *case retention,* which are two important components in the traditional CBR cycle.[1][9] This also means that case recommendation include case reuse and case retention at the same time.

For the second case, the adjustment of demands is the process of demand adaptation that corresponds to problem adaptation. An obvious difference between e-commerce and traditional commerce really lies here; that is, it is relatively difficult for a customer in the traditional commerce to adjust his demand when he is in the market. Usually what he can do is to buy what he sees (BWS). However, in e-commerce, a customer has a much broader space for selecting products. In fact, all available products in the Internet might be searched and selected by any customer if he can access the Internet. In this case, he usually adjusts his demands and tries to get more satisfactory products. Therefore, *problem adaptation* or *demand adaptation* is an important part for a CECS.[b]

After having adjusted the demands, the customer then submits it to the CECS. The CECS will do case retrieval and case recommendation once again. Therefore, the problem submission, case retrieval, case recommendation (including case reuse), and problem

---

a. Case update is an important component of case retention. [1]

b. Notes that demand adaptation can also happen in traditional commerce. Demand adaptation depends on the customer's product selection space.

(demand) adaptation constitutes a cycle. This is the first cyclical process of the CECS, demand adaptation cycle in CECS, which differs from the CBR cycle, and is illustrated as in Fig. 3.

This cycle realizes the following logical model:[10][25]

$$\begin{array}{l} P' \\ P' \sim P \\ \underline{P \to Q} \\ \therefore Q \end{array} \tag{9}$$

It is obvious that model (9) lacks the beauty of similarity. From a technical viewpoint, the realization of (9) in a CECS is a simplification of real word business scenarios.

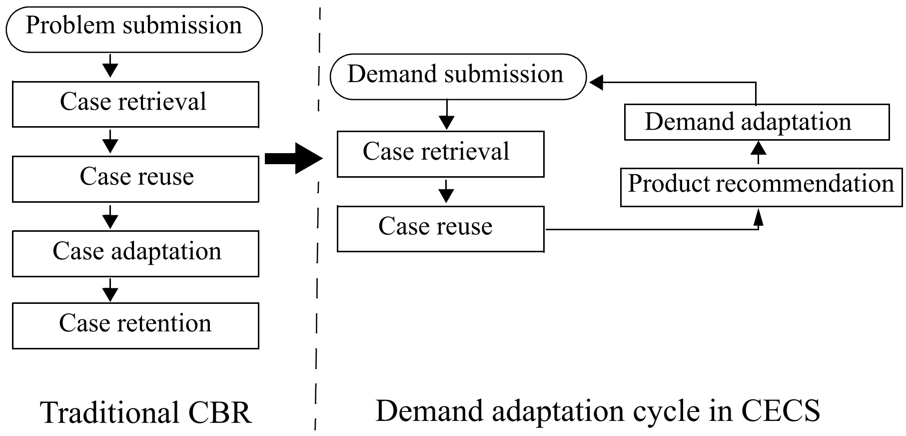

Fig. 3. The first cyclical process in CECS

4.4 A Cyclical Model for Case Adaptation Including Solution Adaptation

Now let us develop the above model naturally: After problem or demand adaptations, the CECS must be aware that the recommended products, for brevity, $q_1, q_2, \ldots, q_m$, may not meet the demands of the customer completely. What the CECS can do under this circumstance is to change the product descriptions (i.e., q'). The process of adjusting the product descriptions is essentially *product adaptation*[a]. It can also be called *solution adaptation* in terms of CBR.

In the product adaptation, the CECS uses its similarity metric mechanism (based on $\approx$, which differs from the case retrieval-oriented similarity metric) to search and retrieve other available products q' (not only from its own case base but also other available case base, in some circumstances). For example, the CECS can use its special search engine to search for the most similar product q'_i to q_i for any $i = 1, 2, \ldots, m$ in the Internet or the case base of its "partner";[a] that is,

$$C(q') = \{q'_i | i = 1, 2, \ldots, m;\ q'_i \approx q_i\}^{b} \tag{10}$$

where, as mentioned, $q_1, q_2, \ldots, q_m$ are prior recommended products. At this point the CECS passes $q'_1, q'_2, \ldots, q'_m$ to the user interface as *new recommended products* based on product adaptation.

After the customer obtains new recommended product descriptions from the CECS, he will further evaluate the recommended products and select one of the following:

1. Accept one of the recommended products, q'_k, and order it, where $1 \le k \le m$
2. Hope that the CECS performs product adaptation in order to pass him revised recommended products
3. Adjust and then send his demand descriptions p' to the CECS for product adaptation in order to pass him revised recommended products
4. Refuse the recommended products and leave the CECS.

---

a. Product configuration can be considered as a kind of product adaptation.

a. This idea is based on the function of meta-search engine, which can search an item from not only its own data base but also other available data bases.

b. This is only one strategy for selecting new recommended products. One can try to use other strategies to do it. For example, $C(q') = \{q'_i | i = 1, 2, \ldots, m;\ q'_i \approx q_1\}$, etc.

Among these four cases only the first three require to study further.

If he selects the first case, then the product adaptation based product recommendation of the CECS has been successful. The CECS will *retain* the new case, (p', q'_k) in the case base of the CECS.

However, if the customer selects the second case, then the CECS must perform product adaptation based product recommendation once again until the customer is satisfied or selects the fourth case. This means that the solution adaptation or product adaptation is also a cyclical process. More specifically, case retrieval, case recommendation, and solution (product) adaptation constitute another cycle, as shown in Fig. 4

This is the second cyclical process of the CECS, product adaptation cycle in CECS, which is different from the CBR cycle, and also results from the application of CBR in e-commerce, and it realizes the following logical model:[11][23]

$$\frac{\begin{array}{l} P \\ P \rightarrow Q \\ Q \approx Q' \end{array}}{\therefore Q'} \tag{11}$$

It is obvious that model (11) also lacks the beauty of similarity from a theoretical viewpoint. From a technical viewpoint, the realization of (11) in a CECS is a simplification of real word business scenarios which include product recommendation, product adaptation and product configuration.

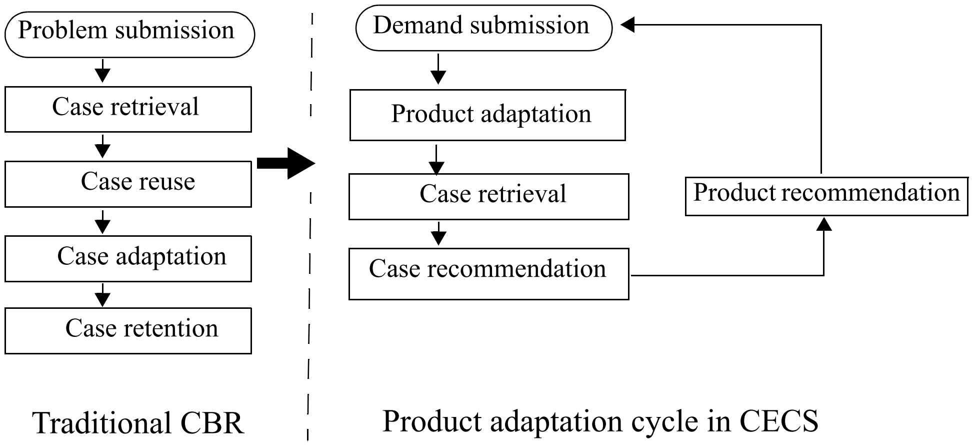

Fig. 4. The second cyclical process in CECS

4.5 Further Comments to Problem Adaptation and Solution Adaptation

Based on above discussion, case adaptation[26] should be decomposed into *problem adaptation* and *solution adaptation,* which correspond to *demand adaptation* of the customer and *product adaptation* of the CECS respectively in e-commerce. Demand adaptation and product adaptation will further be examined in CBR-based negotiation in Section 5. Because a case consists of a problem description and solution description, the adaptation in the CBR catalog system[18] is, in essence, problem adaptation or demand adaptation. By problem adaptation, the customer changes or adjusts his demands to suit the available product (descriptions). The philosophy behind problem adaptation is that in many cases, the customer requirements are not crisp but flexible.

It should be noted that in a more general sense, demand adaptation, which corresponds to problem adaptation in terms of CBR, belongs to the customers' requirements acquisition phase of e-commerce.[3] However, the former is not the same as the latter, although some research (e.g. Ref. [3]) has drawn attention to the latter. The reason is that the customers' requirements acquisition is a more general concept, which corresponds to knowledge acquisition in AI,[25] while problem adaptation is customer's requirement adjustment, a response from a customer to product recommendation of the CECS.

By solution adaptation, the CECS tries to change the product configuration in order to meet the requirements of the customers.

Problem adaptation and solution adaptation together constitute the case adaptation stage in the CBR cycle. Therefore, the decomposition of case into problem and solution is also necessary for case adaptation. Both problem adaptation and solution adaptation should be combined in a CBR system (for example, in CECS) to provide powerful intelligent sales support for e-commerce.

4.6 A Cyclical Model for Problem Adaptation and Solution Adaptation

Now, let us return to the analysis of the decision-making of the customer after he obtains the recommended products as discussed in Section 4.4.

If he selects the third case, this means that the customer and the CECS are negotiating with each other and trying to reach a compromise. This process is said to be *product negotiation* (more generally, it can be called *case negotiation* in terms of CBR).[24][25] This also implies that the case-based product negotiation requires both problem (demand) adaptation and solution (product) adaptation. It is easy to understand that the product negotiation is a cyclical process. Therefore, the integration of problem adaptation and solution (product) adaptation is a cyclical process. This is the third cycle of the CECS, product negotiation cycle in CECS, which differs from the CBR-cycle, and also results from the application of CBR in e-commerce, as shown in Fig. 5.

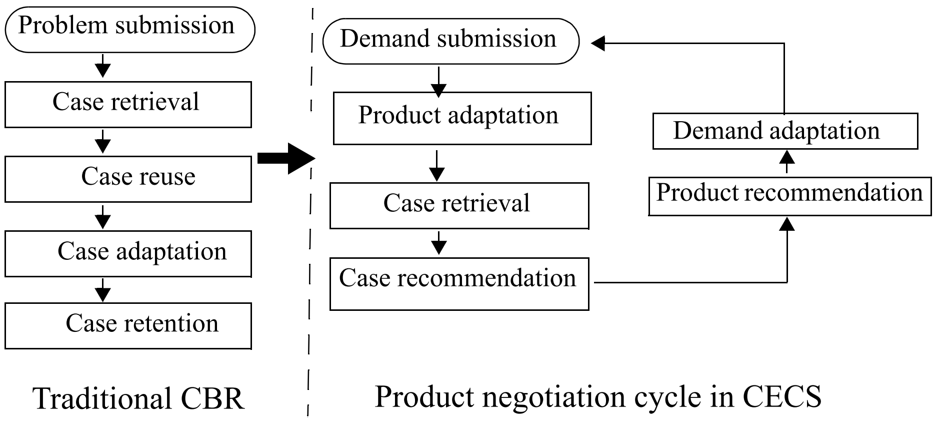

Fig. 5. The third cyclical process in CECS

This cycle realizes the following logical model:[11][25]

$$\begin{array}{c} P' \\ P' \sim P \\ P \rightarrow Q \\ \underline{Q \approx Q'} \\ \therefore Q' \end{array} \tag{12}$$

Strictly speaking, model (12) is an integration of model (10) and (11) taking into account product negotiation. It is interesting to note that this model is identical to the logical model (17) for the CBR cycle given in Ref. [11]. In other words, the above logical model (12) corresponds to case selection, case retrieval, case recommendation, case adaptation, case negotiation and case retention, which once again correspond to application areas in e-commerce such as demand adaptation, product selection, product retrieval, product recommendation, product adaptation, and product negotiation, The detailed correspondence relationship between the activities of the traditional CBR cycle and the activities in the CECS is shown in Fig. 6. This also explains why CBR can be used in e-commerce and where CBR

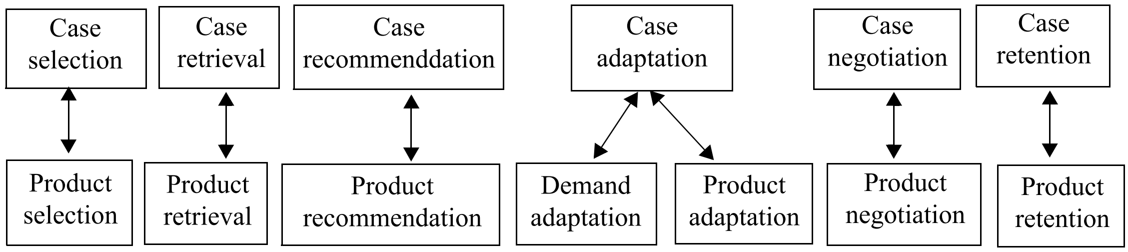

Fig. 6. Correspondence between the CBR cycle and the CECS

can be applied in e-commerce.

After product negotiation, the customer accepts the recommended product, q', in (p', q') with his revised demand p'. (p', q') is a successful selling case which has not been in the case base. Therefore, the CECS will *retain* the new case, (p', q'), in the case base of the CECS. Noted that this idea is also different from that in Ref. [31], where a retain phase doesn't take place because a successful selling of a product will not lead to an additional product.

The previous proposed three cyclical processes in the CECS can be integrated in a system architecture, as shown in Fig. 7.

It should be noted that the proposed model here is not only to model the main activities in e-commerce with CBR, but also to explain why CBR can be applied to product recommendation, product configuration, and product negotiation as well as to give a new insight into traditional CBR; that is, decomposition of case adaptation into problem adaptation and solution adaptation. In this way, case-based product recommendation, case-based product configuration, and case-based product negotiation can be treated in a unified way.[25] In other words, case-based product recommendation, case-based product configuration, and case-based product negotiation are all subsystems of the CECS, and intelligent sales support with CBR[4] can be considered a special case of the CECS.

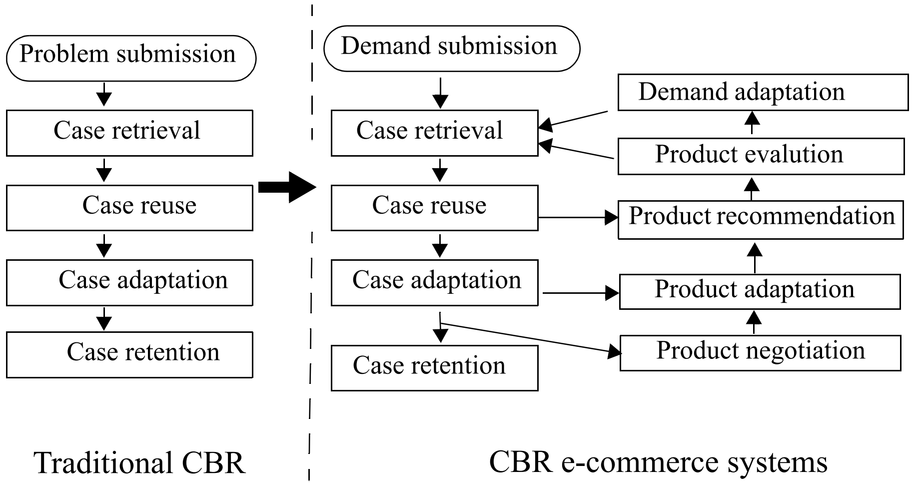

Fig. 7. An integrated architecture for the CECS

5. Case-Based Negotiation

Case-based negotiation has drawn interest in the CBR community,[24][25][30][32] for example, Zhang and Wong [32] examined a negotiation framework that is called Case-Based Negotiation (CBN). The CBN applies CBR to represent and reuse previous negotiation experiences. This section will examine case-based negotiation from an original viewpoint by arguing it as an integration of problem adaptation and solution adaptation in more detail. This integration or combination is necessary for maximizing the customer satisfaction during e-sales.

Negotiation in e-commerce is a process where two parties bargain resources for an intended gain, using tools, and techniques of e-commerce.[20][31] Negotiation is an important part of the selling process not only in traditional commerce but also doing business on the Internet. In order to support customers' purchase in a sufficient way, CECS should possess the ability of negotiating.[31] However, online negotiation is a process with relatively little support to date become of the complexity of the negotiation process, which depends on the complexity of the product or service being negotiated.

It is assumed that a conflict is in the price of a commodity with *n* attributes. Then, the existing different approaches for negotiation in e-commerce can be classified into two classes: a cooperative approach and a competitive approach.[12] Competitive negotiation takes place if there is at least a conflict of interests between the buyer and seller. Consequently, there will not be more collaboration than necessary between the buyer and the seller to solve the negotiation problem, although cooperative negotiation tries to get as much collaboration as possible between the two negotiating parties. However, both approaches present two extremes on a continuum of possible underlying problems. In practice, negotiation during the business bargaining process is a compromising negotiation, which lies in between cooperative negotiation and competitive negotiation.

During the sales process, the customers are navigating through the available products and searching for a product that meets their demands.[31] Some demands are known in advance and additional ones may be discovered during the navigation in the product space. Some demands are *fixed* and must be fulfilled by the product and other demands are more or less *weak or flexible*. Generally speaking, the customer's satisfaction is maximal if the modification of his weak and hard demands is minimal and he finds his product as quickly as possible. The main task of a CBR-based negotiation system is to identify these demands in cooperation with the customers, perform demand adaptation, and then to perform product adaptation to meet these demands. For demand adaptation, the CBR-based negotiation system might suggest or even add some new demands or modify some weak demands for the purpose of finding an appropriate product. For product adaptation, it is also possible for the CBR-based negotiation system to modify existing products during product (solution) adaptation to meet the customer's demands. Such demand adaptation and product adaptation might be performed many times in order to maximize customer's satisfaction. Therefore the main task for the CBR-based negotiation system during the negotiation process is an *iterative process of demand adaptation* and *product adaptation*[a], the iterative modifications of demands and/or the product.

It should be noted that in investigating e-negotiation from a CBR viewpoint, we find that there is no stage in both the traditional CBR cycle[1] and improved CBR cycle[9] which corresponds to negotiation. However, after decomposing case adaptation into problem adaptation and solution adaptation, we find the relationship between e-negotiation and problem adaptation as well solution adaptation. The iterative process of problem adaptation and solution adaptation corresponds to the bargaining process in e-negotiation. Therefore, in investigating e-commerce applications one can improve the understanding of CBR.

Finally, case-based negotiation is a fruitful area of work and there is much to be done. For example, meta-information about the case base might be useful in successful bargaining in a negotiation, which will be examined in future work.

6. Concluding Remarks

This paper examined new issues resulted from applying CBR in e-commerce and gave a new insight into the traditional CBR-cycle. The main results can be summarized as follows:

- Decomposition of case adaptation into problem (demand) adaptation and solution (product) adaptation is a necessity for applying CBR in e-commerce

a. Noted that the mentioned user demands corresponds to the *problem descriptions*, while the products to the *product descriptions* in the *sale case.* [25]

- Solution (product) adaptation is a process following case recommendation providing that the demand adaptation based case recommendation is not successful
- Case (product) negotiation is an integration of problem (demand) adaptation and solution (product) adaptation
- Product adaptation based case recommendation and demand adaptation based case recommendation together can improve the satisfaction of the customer.

Furthermore, this work has shown in a unified way that CBR technology can be applied to intelligent sales support, product search, product selection, product recommendation, product configuration, and product negotiation in e-commerce based on the proposed unified logical model for CBR-based e-commerce systems. Applying CBR in e-commerce has been treated here at three levels: logical level, CBR-level, and e-commerce level. This is a new attempt in both CBR and e-commerce.

From a CBR viewpoint, customer demand adaptation can be considered as problem adaptation, while product (or goods) adaptation can be taken as solution adaptation. Customer demand adaptation and product adaptation play an important role in e-commerce and lead to two different categories of product recommendation: customer-oriented product recommendation and system-based product recommendation. Customer demand adaptation is suited to the business situation that the customer can adjust his demands or problem descriptions with patience. Product adaptation can serve the business situation in which the required products or goods should be tuned to best satisfy the requirements of the customer. In practice, both problem adaptation and solution adaptation should be combined in a CBR system to provide powerful intelligent support for e-commerce.

Beside those mentioned in this paper, some issues in the following fields can be considered to lead to new advances in CBR in e-commerce: auctions, brokering/negotiation, customer relationship management, and supply chain management (see http://www.ics.uci.edu/~burke/research/cbrec/cfp.html). Further, the effective application of CBR in e-commerce will certainly be facilitated through multiagent systems, because intelligent agents will play an important role in e-commerce, just as human agents have done and are doing in traditional commerce.

Bibliography

1. Aamodt A, Plaza E. Case-based reasoning: Foundational issues, methodological Variations, and system approaches, *Artificial Intelligence Communications*, IOS Press, 7(1) 1994. pp 39-59
2. Bergmann R, Schmitt S, Stahl A. Intelligent customer support for product selection with case-based reasoning, E-commerce and Intelligent Methods, Physica-Verlag, 2002. pp 322-341
3. Bergmann R, Cunningham P: Acquiring customers' requirements in electronic commerce. Artificial Intelligence Review, Vol 18 (3-4) 2002 pp 163-193
4. Cunningham P. Intelligent systems in electronic commerce. Invited talk on ICCBR-99, Germany; 1999. URL: http://www.iccbr.org/iccbr99/. 1999
5. Cunningham P, Bergmann R, Schmitt S, Traphoener R, Breen S, Smyth B: WEBSELL: Intelligent Sales Assistants for the World Wide Web, Trinity College Dublin, Computer Science, Technical Report, TCD-CS-2000-42, 2000. at URL: http://www.cs.tcd.ie/publications/tech-reports/reports.00/TCD-CS-2000-42.pdf
6. Cunningham P, Bergmann R, Schmitt S, Breen S, Smyth B, Traphoener: Intelligent support for online sales: The Websell experience, 2001. at URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc6.pdf
7. Dubois D, Prade H. Similarity-Based Approximate Reasoning. *Computational Intelligence: Imitating life*, New York: IEEE Press; 1994. pp 69-80

8. Finnie G, Sun Z. Similarity and metrics in case-based reasoning. *Int J of Intelligent System,* 17(3) 2002. pp 273-287

9. Finnie G, Sun Z. R^5 model for case-based reasoning. *Knowledge-based Systems*, 16 (1) 2003. pp 59-65

10. Finnie, G, Sun, Z. A knowledge-based model of multiagent CBR systems. In Mohammadian (Ed.) *Proc of Int Conf on Intelligent Agents, Web Technologies and Internet Commerce (IAWTIC 2003)*, Vienna, Austria, 2003. pp 494-503

11. Finnie G and Sun Z. A logical foundation for the CBR cycle. *Int J of Intelligent Systems, 18 (4)* 2003. pp 367-382

12. Guttman RH, Moukas AG, Maes P. Agent-mentioned electronic commerce: A survey. *Knowledge Engineering Review*, 13(2) 1998. pp 147-159. URL: http://ecommerce.media.mit.edu

13. Hayes C, Cunningham P. Shaping a CBR view with XML. In: *Proc. of Int Conf on CBR (ICCBR'99*) Munich, Germany, July 2000. pp 468-481

14. Hayes C, Cunningham P, Doyle M. Distributed CBR using XML. In: *Workshop for Intelligent Systems and Electronic Commerce* (within German Conference on Artificial Intelligence (KI'98)). September 15-17 1998

15. Inakoshi H, Okamoto S, Ohta Y, Yugami N. Effective decision support for product configuration by using CBR. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws3/ws3toc4.pdf

16. Leake D. *Case-Based Reasoning: Experiences, Lessons & Future Direction*. Menlo Park, California: AAAI Press / MIT Press; 1996

17. Leake DB, Plaza E (eds): *Case-Based Reasoning Research and Development. Proc 2nd Int Conf on CBR, IC-CBR-97*. Providence, USA, Jul. 1997, Springer; 1997

18. Lenz M, Bartsch-Spörl B, Burkhard HD, Wess S (eds): *Case-Based Reasoning Technology, from Foundations to Applications*, Berlin: Springer; 1998

19. Liu JNK, Leung TTS. A Web-based CBR agent for financial forecasting. URL: http://www.aic.nrl.navy.mil/papers/2001/AIC-01-003/ws5/ws5toc11.pdf. 2001

20. Novais P, Brito L, Neves J. Agreement in virtual marketplaces with CBR-supported negotiation. In: *Proc. PAAM 2000- 5th Int Conf on the Practical Application of Intelligent Agents and Multiagents*. Manchester; 2000. pp 203-206

21. Schmitt S, Bergmann R: Applying case-based reasoning technology for product selection and customization in electronic commerce environments. In: *12th International Bled Electronic Commerce Confenrece*, Bled, Slovenia, June 7-9, 1999

22. Smyth B, Cunningham P (eds): *Advances in Case-based Reasoning*. LNAI 1488. Berlin: Springer-Verlag; 1998

23. Sun Z, Finnie G. Multiagent brokerage with CBR. In: *Proc Australasian Conf on Information System (ACIS'01)*. Coffs Harbour, Australia; 4-7 December 2001. pp 635-644

24. Sun Z, Finnie G. Intelligent techniques in e-commerce: A CBR approach. In: Faucher C, Jain L, Ichalkaranje N (eds): *Innovations in Knowledge Engineering*. Adelaide: Advanced Knowledge International; 2003. pp 245-280

25. Sun Z. *Case Based Reasoning in E-Commerce*, PhD Thesis, School of Information Technology, Bond University, 2002

26. Voss A. Towards a methodology for case adaptation. In: Wahlster W (ed.): *Proc 12th European Conf on Artificial Intelligence (ECAI'96)*, John Wiley and Sons; 1996. pp 147-151

27. Watson I. An introduction to case-based reasoning. In: Waston I (ed.): *Progress in Case-based reasoning*. Berlin: Springer; 1995. pp 3-16

28. Watson I. *Applying Case-Based Reasoning: Techniques for Enterprise Systems*. San Francisco, California: Morgan Kaufmann Publishers; 1997

29. Watson I, Gardingen D. A distributed case-based reasoning application for engineering sales support. In: *Proc 16th Int Joint Conf on Artificial Intelligence (IJCAI-99), Vol.* 1, Morgen Kaufmann Publishers; 1999. pp 600-605

30. Wilke W, Bergmann R, Wess S, Negotiation during intelligent sales support with case-based reasoning, *Proceedings of the 6th German Workshop on Case-Based Reasoning (GWCBR'98)*, 1998, URL: http://wwwagr.informatik.uni-kl.de/index2.html?click=2

31. Wilke W, Lenz M, Wess S. Intelligent sales support with CBR. In: Lenz M, et al. (eds): *Case Based Reasoning Technology: from foundations to applications*. Berlin: Springer LNCS 1400. 1998. pp 91-113

32. Zhang D, Wong W. Using CBR for adaptive negotiation. In: Shi Z, Faltings B, Musen M (eds): *Proc Conf on Intelligent Information Processing (IIP2000)*, Aug. 21-25, 2000 Beijing. pp 428-437.