Fachbereich I – Psychologie
Universität Trier

Diplomarbeit

Conception, Implementation, and Evaluation of a Case-Based System for Sales Support in the Internet

Stephan Weibelzahl
Egbertstr.22
54295 Trier
weibelzahl@cogpsy.uni-trier.de

Presented to

Prof. Dr. Gerhard Weber

and

Prof. Dr. Karl F. Wender

April 1999

Contents

Preface

This thesis originates in a cooperation between tecInno, Kaiserslautern and the University of Trier. Thanks to this constellation many people were involved in the present project. It is a pleasure to thank them for their support.

Special thanks go to Gerhard Weber, my thesis advisor, who generously provided feedback whenever I needed it, and who supported me in every stage of the project.

Thanks to the people at tecInno for answering my silly questions: Thomas Roth-Berghofer, Stefan Wess, Stefan Schulz and many others. Ralph Bergmann, Carl Branting, and Anthony Jameson gave invaluable suggestions on how to improve various aspects of my project. I would like to thank them for their thoughtful advise.

Ein *dickes Lob* muß ich Julia Spaniol aussprechen, die sich durch die gesamte Diplomarbeit quälte und meine gestolperten englischen Sätze in einen richtigen Text verwandelte.

Finally I would like to thank my friend for bearing with my antisocialness during the long hours that I spent thinking about cases and similarities.

St. Weibelzahl

Part I.

Online Sales Support in the Internet

1. Existing Systems

Internet sales are booming. The rate of increase in e-commerce, *the exchange of information, goods, or services through electronic networks* (Wilke, Lenz, & Wess, 1998), is enormous. The investment bank Hambrecht & Quist estimates the volume of business-to-business trade in 2000 around $ 105 billion. This figure is more than 13 times higher than that reached in 1997 (Kosche, 1998).

But the figures suggest a reality that does not truly exist. If you browse today's Internet, you will notice that the quality of online shops still leaves much to be desired. Having to search long lists of poorly presented products, customers experience something akin to being left alone in a warehouse. Purchasing a product online does not provide the customer with any special benefits.

So far only one sales system has been implemented in the Internet: the choose-out-of-a-list system. Thus, Internet shops resemble old-fashioned paper catalogues with regard to their structure and the selection posssibilities they offer. Customers can choose a certain product category and are presented with a list describing each article in detail, sometimes featuring pictures. A typical representative of this type of Internet shop is the Karstadt- shop (*http://www.karstadt.de*). This site is a simply a replica of the company's hard-copy catalogue. For instance, toasters are presented in the *Houshold & Electrical devices* category, where you can find a list of 30 products. The system does not give any advices or hints during the search, so the handling remains the same in comparison to the hard-copy catalogue.

But there are a few examples of systems that try to take advantage of the new network technology. So-called *sales agents* support every step of the interaction between customers and online shops. A sales agent is a computer based system that offers *intelligent sales support* by mediating between customers and products, in a fashion similar to that employed by real salesmen.

With regard to the type of task performed by the sales agent it is possible to distinguish between four systems (Bräuer & Zimmermann, 1997):

- *Electronic catalogues* lead customers to the desired product by supporting their choice. The emphasis of these systems is on finding the best match to the customers' query.
- *Configuration systems* offer the opportunity to specify search criteria and to assemble the product out of disjoint parts.
- *Presentation systems* concentrate on communicating the features of a product through visualization and labeling.
- *Help desk systems* are used in after-sales support, offering help with problems that may arise while using the product.

I will mainly adress electronic catalogues and presentation systems. This will entail looking at the pre-sales and sales processes (Wilke et al., 1998) in products that are ready for purchase and without having to be configured individually.

All of the sales systems described above share a common goal: improving consumer satisfaction. The next two sections will outline several approaches towards achieving this goal.

2. Sales Support from a Data-Processing Perspective

This chapter presents various ways in which sales support can be improved by adapting a data-processing perspective.

Searching and purchasing products on the Internet may be viewed as a simple database-query as show in figure 2.1. A customer is looking for a specific product and therefore sends a query to the sales agent. The sales agent responds to the query by stating whether the requested product is available or not.

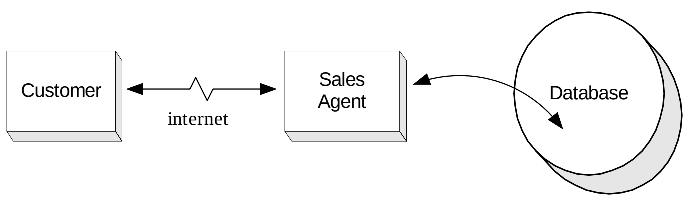

FIGURE 2.1.: Simple model of selling via the Internet

There are several ways in which Internet sales support can be improved from a data-processing point of view. The processes of collecting data, transforming the users' query, and responding to the query can be optimized by reducing the systems' response-time, providing an accurate representation of the domain, and by searching the domain efficiently.

Reducing response time enhances user comfort by minimizing waiting time. However, there is a trade-off between response time and response accuracy: If you aim at reducing response time, you need a small, simple, and manageable representation of the domain. On the other hand, the representation should contain all the information that is necessary for responding with high accuracy to the customer's query. But the more dimensions and attributes are modeled in the representation of the domain, the more difficult it gets to search the problem space within a reasonable amount of time. Elaborate models and algorithms help to deal with this problem.

Without going into detail, it should be pointed out that systems have been developed using a case-based reasoning (CBR) approach, first described by Schank (1986). Part II will introduce the basic principles and techniques employed by this inferential approach.

3. Sales Support from a Psychological Perspective

This chapter outlines some ideas of how psychology might contribute to the improvement of sales agents.

Modeling Internet business as a database query—as illustrated in figure 2.1—obviously neglects all aspects concerning the user as an individual. Psychological theories, however, take into account the fact that customers have different needs. Moreover, these vary not only between individuals, but also across situations. In this context, various factors have received attention: the human-computer interface, the psychology of marketing, and user modeling as a way to enhance individual sales support.
Each of these factors will be considered in turn.

3.1. Ergonomics and Marketing

Recent research in the field of human-computer-interaction (HCI) has made great advances defining criteria that define interface quality.

The main goal is to make the dialog between user and computer as smooth as possible. Designing user-friendly displays and supplying the user with maximal control are important steps towards improving the interface (Ulich, 1994).
It makes great sense to apply this new knowledge in the area of Internet sales support.

Industrial psychologists have devoted many efforts identifying conditions that foster success in selling products. The customer's perception of a product and the emotional response elicited by that product have been found to be powerful predictors of sales success.

Of even more potential impact on the design of sales agents are theories of consumer decision-making. These theories try to explain who decides to buy which product in which situation, and why (Kroeber-Riel & Weinberg, 1996). How can we predict the behavior of customers? Which inherent attributes of a product determine the selection? Again, the insights gained from this perspective lend themselves to an integration with existing views on online customer support. Chapter 7 will further explore this subject.

3.2. User Modeling

While HCI research and industrial psychology focus on general principles of human behavior, cognitive psychology has gone in a different direction to improve sales support. The basic assumption is that it is possible to construct a model of the user that will help provide individual support. The strategy is to collect information about the user and to use these data to adjust support. Thus each user interacts with a system that is custom-tailored to meet his or her meets.

There is a wide range of different user models that vary in the tasks they can perform. This chapter will describe a few exemplary approaches from the field of sales support. Adaptation to the user profile can take effect in various ways, ranging from how information is presented to more complex tasks, such as predicting the users' actions, or adapting to the needs of the user.

3.2.1. Adapting Information-Presentation

Often users are overwhelmed by the multitude of details about the products that are presented to them. They have trouble filtering out the important information and quickly get confused and frustrated. But how can a system know which information is relevant for a specific customer?

To solve this problem Rössel (1997) introduced a system that records the users actions in an introduction phase, based on which the user is assigned to a certain category (Lödel, 1994) that constitutes a combination of six dimensions listed in table 3.1. These six dimensions permit the system to dicriminate between $2 \times 3 \times 3 \times 2 \times 2 \times 4 = 288$ different user models.

Equipped with this background knowledge the system is able take the user on a *guided tour* through a branch catalogue. Depending on the user model at hand the system selects suitable information, while continually updating the user model. E.g., a user seeking information about catalytic converters will view an individual selection of pictures, films, texts, etc., about this topic that is custom-tailored to fit his knowledge,

TABLE 3.1.: Dimensions and categories of the user model in Rössel's system. Adapted from Rössel (1998).

Dimensions	**categories**			
goal	overview	details		
time budget	low	middle	high	
branch-knowledge	novice	advanced	expert	
system-knowledge	novice	experienced		
cognitive style	holistic	serial		
preferred style of presentation	intellectual	photographic	cinematic	auditory

goals, and his time budget. After this the user will be treated as an expert of catalytic converters, such that more detailed and specialized information can be presented the next time.

Another approach to optimizing the presentation of information was described by Linden, Hanks, and Lesh (1997). The so-called *Automated Travel Assistant* is able to build flight itineraries interactively. Besides several other interesting features that will be described later, this system uses a special way to present the flight offers to the customer. After having the user define her preferences (e.g., date, destination, departure time, airline), a set of suitable itineraries sorted into three categories is displayed to the user. An example is given in figure 3.1.

The first category is labeled *Best Trips* and contains those three itineraries that best match the user's preferences. In the remaining two categories the system suggests *extreme options* labeled *Cheapest Trip* and *Best Nonstop*.

To allow the customer to get an idea ATA not only presents the customer with the choices that optimally match all of her preferences but also with options that differentially weigh those preferences. So she gets an impression of the range of options. Moreover the model of preferences is updated if she shows interest in one of the *extreme options*.

Again—as we have seen before—the user model enables the system to improve its performance in a context-sensitive fashion. Jameson, Schäfer, Simons, and Weis (1995) provide an overview of existing information-filtering systems that employ similar user models.

This small example illustrates some of the advantages of user models. By evaluating the

Best Trips
San Jose, CA (SJC) →Philadelphia, PA (PHL) →San Jose, CA (SJC)
{American} $503.00

San Jose, CA (SJC) →Philadelphia, PA (PHL) →San Jose, CA (SJC)
{USAir} $523.00

San Jose, CA (SJC) →Philadelphia, PA (PHL) →San Jose, CA (SJC)
{American} $503.00

Cheapest Trip
San Jose, CA (SJC) →Philadelphia, PA (PHL) →San Jose, CA (SJC)
{USAir, Reno Air, United} $353.00

Best Nonstop
None

FIGURE 3.1.: Trips displayed by the ATA after the initial query for a round-trip from San Jose to Philadelphia. Each trip can be expanded to show information about the flights. Adapted from Linden et al. (1997).

information it obtains from the user, the system does some of the work (in this case, information selection) that the user would normally have to do. This allows the user to concentrate on the primary task at hand: making a choice between various alternatives.

3.2.2. Adapting Actions

Of course, the application of user adaptation models is not limited to the presentation of information. More complex tasks, such as the selection of alternatives, can also be performed by means of user modeling.

E.g., the *ATA* described in the previous section is able to perform a complete sales interaction by employing a *candidate/critique* model. After presenting some *candidate solutions* the user *critiques* those solutions. If the user critiques one of the *extreme* solutions by noting favorable or unfavorable characteristics, this information is then used by the system to refine the user model.

If, for instance, the user states that the cheapest flight is attractive but that its depature time is too late, the system infers a high preference for low price and an intermediate preference for optimal departure time, and will select new alternatives correspondingly.

The interaction eventually leads to an acceptable solution. Adaptation in this context involves not only the ability of the system to respond to a single query but also its use of implicit information revealed in the interaction, especially in the user's critique of candidate solutions.

Note that the refinement process is performed more efficiently if extreme options are presented because they enable the system to learn how far the user is willing to depart from her initially stated preferences, and which preferences receive priority. Thus, the user model can be updated more accurately in the desired direction.

Taking this principle one step further, systems can even select questions that will yield evidence about the user's preferences, which makes the interaction more flexible and comfortable for the user. This strategy is used in PRACMA (Jameson et al., 1995). When interacting with a customer this system selects the next dialog move based on its knowledge about the user.

This is illustrated by the following scenario: In the domain of used car sales a customer might express a preference for inexpensive cars, that are not too old. The system might offer a certain car that meets these criteria. Now it has to decide about the next action, taking into consideration both its current user model and its own dialog plans and strategies. E.g., if PRACMA finds that mentioning a certain attribute will likely produce an *evaluation shift* in the customer, then it will continue with presenting the product, instead of asking about further criteria. Section 3.2.3 will explain how an evaluation shift is predicted computationally.

These examples demonstrate how a system can adapt its behavior to the user by incorporating a user model that helps guide the selection of actions.

3.2.3. Adapting to Needs

All systems that have been mentioned so far are useful improving sales support. However, from a psychological point of view they are deficient in at least one aspect: the interaction feels artificial and mechanical. Imagine a customer talking to a travel agent:

> *Can you offer me a vacation home in France that is available from the 4th to the 18th of august? The ocean should be no more than 2.5 km away and there must be a microwave oven in the kitchen and SatTV in the living room.*

This might sound funny but it is in fact very close to the degree of detail required by some online stores, especially those selling more complex products such as vacation homes or automobiles. Clients have to go through long lists of specific product attributes, which often leads to problems (Mehlmann, Landvogt, Jameson, Rist, & Schäfer, 1998):

- The customers have to decide about a lot of detail, rating attribute by attribute. This is rather uneconomic and can be frustrating.
- Attribute lists are written in an industrial jargon. They typically contain exact numeric information (e.g., lengths, weights, material, article numbers). These descriptions vastly differ from what the customers have in mind about the product. They would prefer to describe what they want—as they do in a real store—in their own words.
- On the other hand, the customers' statements often lack in precision and completeness. Since they are not familiar with the producers' technical terms, miscommunications can be expected to occur frequently. Customers might misinterpret a dimension proposed for rating or they might just refuse to respond.

Table 3.2 illustrates the gap between the two levels of description. It presents the customers' descriptions in terms of motives and preferences on the one hand, and the producers' technical descriptions on the other hand.

TABLE 3.2.: Differences between the description of the product provided by the producer and the language of the customers

Description of product	**Language of customers**
exact	inaccurate
complete	incomplete
concrete	abstractedly
detailed	comprehensive
attributes	needs and preferences
misinterpreted by customer	misinterpreted by sales agent

One step towards improving adaptation to the customer needs was taken by Rosewitz and Timm (1997). They introduced a system that enhances the typical electronic catalogue in two steps.

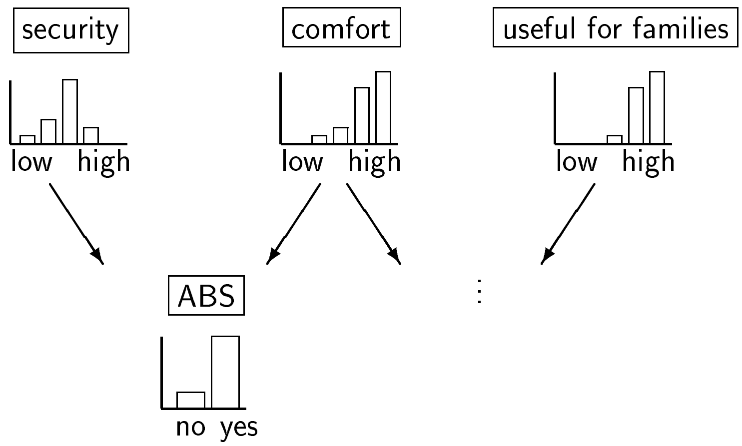

FIGURE 3.2.: Part of an exemplary Bayesian Network in the domain of cars

First, the product description is extended. In addition to usual attributes like price and features, expert ratings of every product in terms of comfort, quality, reliability, price segment, and other relevant categories are presented.

In a second step performed during the interaction the system constructs a user model by recording patterns of purchase for every customer. E.g., the system might notice that a customer always buys products from a lower price segment but rejects products of very low comfort. When the customer looks for another product, say a TV set, the system automatically selects an offer from the database that meets the requirement *low price* and *middle or high comfort*, though the customer never explicitly expressed this need.

This feature of the system sgnificantly improves the interaction, since it is able to present products that are probably closer to the customer's needs. But the customer still has to deal with highly technical product descriptions.

Mehlmann et al. (1998) present an automobile-selling system that overcomes this difficulty. The developers consulted marketing research to identify the most important abstract attributes and dimensions that people apply when buying a car. Typical examples are *safety* and *family-friendliness*. Then, the developers had experts rate every attribute of the product in relation to these abstract attributes. For instance, *ABS* might be highly related to the attribute *safety* but is irrelevant for the feature *family-friendliness*, as shown in figure 3.2. On the basis of this information a Bayesian Network is built that predicts how a customer with specific needs will evaluate a certain car.

A Bayesian Network describes each attribute in terms of a probability distribution. An attribute's distribution is computed by predicting its probability, given the distribution of its related nodes (i.e. attributes). E.g., the probability of *ABS* is a function of the imaginary customer's desire for *safety* and, say, *comfort*. If the status of an attribute becomes certain, since the user has made an explicit statement, then the status of low-level attributes can be predicted more precisely. Note that expert knowledge and implicit assumptions about the domain are still required to build the system. This paper will present a system that these not require such knowledge.

As we have seen user models can help systems to adapt to the individual needs of a customer. The integration of user models and inference techniques is capable of translating between the customer's language and the technical description of a product.

Part II.

Case-Based Reasoning

In the previous chapters case-based reasoning was mentioned several times described as an intelligent method of inference. What is CBR? How does it work? What kinds of problems can be solved by CBR? What are the strengths and weaknesses of this technique? This part will answer these questions by giving an overview of CBR.

4. Foundations

Until 1986 the General Problem Solver (GPS) represented the basic idea of artificial intelligence: a system equipped with an infinite number of rules can solve every problem. The more rules and the higher the degree of elaboration of those rules, the more intelligent is the system.

Schank (1986) introduced a new idea by positing that human reasoning is mainly built on the use of experience:

> *Case-based reasoning is the essence of how human reasoning works. People reason from experience. They use their own experience if they have a relevant one, or they make use of the experience of others to the extent that they can obtain information about such experiences. An individual's knowledge is the collection of experiences that he has had or that he has heard about* (Riesbeck & Schank, 1989, page 7).

According to this view all knowledge is derived from so-called cases. *A case is a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of a reasoner* (Kolodner, 1993). On the basis of such a representation a case-based reasoner can be built that solves problems by adapting solutions or strategies that have been used to solve other problems.

The CBR-Cycle (figure 4.1) describes the structure of a case-based reasoner. The solution of a problem begins with generating a new case. In a retrieval phase, the general knowledge containing all previous cases is searched for a similar case. But the retrieved case probably does not match the problem case perfectly. The structure of the retrieved case might deviate somewhat or its attributes might be different. Thus, the retrieved case must be adapted to yield an aedequate solution to the new case. A revision phase then tests whether the suggested solution really works. Eventually a set of repair techniques are applied to find best possible solution. Thus, two results are obtained: a solution to the problem, as well as a new experience. Both of them together represent a new case, which is added to the existing knowledge base.

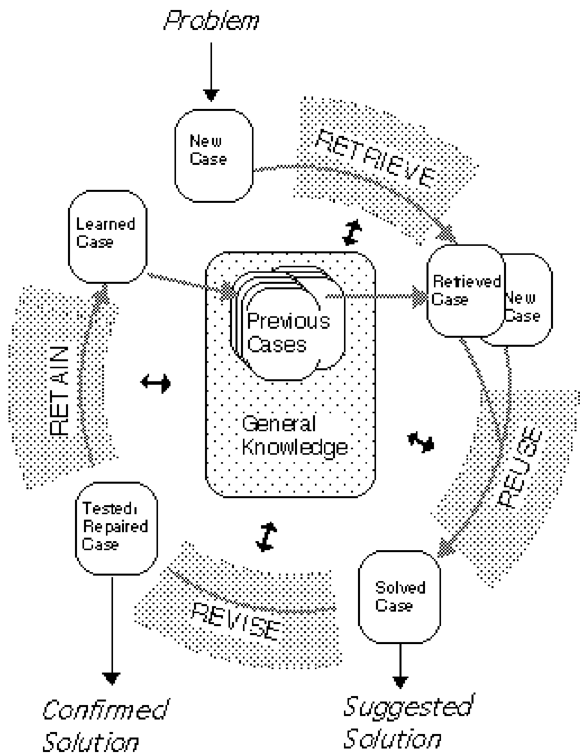

FIGURE 4.1.: The CBR-Cycle adapted from Aamodt and Plaza (1994)

An important problem faced by this theory is the difficulty of defining similarity between cases. Which case has the potential to present a solution to the problem? What criteria are necessary to adequately describe a new problem? The authors' answer to these questions are elaborate domain models that describe the structure of cases and similarity measures that determine relations between cases.

Four components are needed to build a case-based system. They are called *knowledge containers* (Richter, 1998):

1. The *vocabulary* used in the domain.
2. The *similarity measures* defining similarity between cases. While *local similarity* refers to case attributes, *global similarity* describes the total degree of overlap between cases.
3. The *case base*, which means the central knowledge base, that contains all solved cases.
4. The *solution transformation* that describes which adaptations of a case are permissible.

Intelligent systems that solve problems via case-based reasoning have been implemented in a wide range of domains such as legal reasoning, medicine, tutoring, design, and text parsing. There have also been applications in the area of sales support.

5. Case-Based Reasoning in Sales Support

CBR is used in sales support because some of its features are very helpful in building sales agents.

As described in chapter 2 Internet sales can be viewed as a database query. CBR warrants intelligent database access by applying its knowledge base to the problem. A query sent to the database is no longer interpreted word by word but instead it is treated as a problem that can be solved by a similar case. The following example will illustrate this principle. A customer looking for a color TV wants to spend no more than 400 DM. The formal representation of this query looks like the left half of figure 5.1.

query	
product	ColorTV
price	400 DM

⇒

solution	
product	ColorTV
price	401 DM
screen	70 cm
color	black

FIGURE 5.1.: Example of a query sent to a sales agent

Using a word by word interpretation the sales agent might respond that there is no product in the assortment that conforms to these restrictions. Thereupon the customer will probably leave the shop. However, if there is a TV that costs exactly 401 DM, a so-called *near miss* occured. An intelligent sales agent would offer this product because it searches for those products that are the most similar to the query. The customer would probably find this price acceptable although it does not exactly match his initial price criterion.

The opposite scenario could also happen. The customer may find that there is a long list of products that conform to his criteria. Again, he will probably not find this very useful, because it forces him to go through a lengthy selection process. In this scenario, a case-based sales agent would only present a small subset of products. Again, it retrieves only those solutions that are most similar to the query.

This example illustrates one of the major advantages of CBR in sales agents. The customer benefits from the system's knowledge about the domain. Near misses or overwhelmingly long lists of hits can be avoided more easily.

As mentioned before, the CBR approach benefits from the advantages of the data-processing perspective. User modeling has so far not been applied to CBR. Clearly, it would render the CBR approach even more powerful.

Several questions arise with regard to the integration of CBR and user modeling. Can user models be treated as simple cases? Which aspects of the users should be modeled? How can we represent the customer domain within CBR? I will try to answer these questions by conceptualizing a CBR sales agent in the travel domain that features user modeling.

Part III.

Conception and Implementation

6. Problem Specification

We have seen that CBR and user modeling can both contribute to the improvement of sales support on the Internet. A system will be implemented that exemplifies an integration of these techniques. First, several hypotheses are introduced as motivation for the project:

A sales agent can incorporate case-based reasoning and user modeling. It is possible to build a system that is grounded in CBR, yet is capable of providing individual support via user modeling. An exemple from the travel domain will serve to demonstrate how such a combination might work.

The vacation home domain was chosen because due to the complexity of the product, travel agencies have traditionally provided little sales support in this domain. Few travel agents are able to provide detailed information on individual vacation homes. So the customers receive a catalogue and have no choice but to select a home on their own. An intelligent sales agent would be able to guide the customers during this search.

Case-based reasoning is compatible with user models. The user models in such a system are constructed, processed, and utilized in many different ways. CBR is able to perform this job and can even simplify the coordination of different user models.

User modeling improves individual support. User models help improve sales support by providing information necessary for individualized service.

Customers who receive support from a case-based sales agent are more satisfied. In an evaluation phase we will establish measures that allow us to compare different systems. It is expected that customer satisfaction is a function of the quality of sales support. The proposed sales agent will be evaluated against existing Internet sales approaches, such as a choose-out-of-a-list and a traditional CBR system.

The case-based sales agent is implemented in three steps. First, the dimensions that predict customer behavior must be explored. Second, these findings must be operationalized and implemented in a CBR system. Third, a CBR algorithm must be developed to ensure the adequacy of the retrieval and a learning mechanisms employed by the system.

7. Theories of Customer Behavior

The primary goal of theories of customer behavior is to predict the customer's decision when faced with a choice of products. Several approaches have been proposed, ranging from simple economic to more complex psychological theories. Only a few psychological approaches are considered in this chapter.

7.1. Relevant Dimensions of Customer Behavior

Early studies on customer behavior aimed at classifying consumers into types (Bergler, 1987). The idea was that each individual falling under one of these categories would behave in the same way, i.e., choose the same product. This approach was not very successful, since predictions based on an individual's membership in only three or four categories are not very reliable. A more sophisticated approach consisted of characterizing customers along multiple continuous dimensions. This procedure is called market segmentation (Bergler, 1987).

Some commonly used variables for segmentation are demographic characteristics (e.g., age, sex) and personality traits (e.g., extraversion). However, experiments using demographic market segmentation and personality segmentation demonstrated the limits of these techniques. While demographic and personality variables might predict which product categories the customer chooses, they cannot predict which specific product the customer will choose. E.g., demographic variables might be able to predict whether a person smokes cigarettes or not. But these variables give little cues as to which cigarette brand the person prefers (Bergler, 1987).

Psychodemographic segmentation provides more precise measures, as it takes psychological factors into consideration. It is possible to distinguish at least four different types of psychodemographic segmentation:

- *Attitude segmentation* separates groups that share similar attitudes or motives.

E.g., people who would agree to a statement like "We must protect nature" are more likely to prefer to organic fruit and vegetables.

- In *expected value segmentation* those individuals are assigned to one group that share the same expectation about the benefits of buying a specific product. E.g., people might differ in what they expect to gain from continuing education course (Siebert, 1980). While some may hope to have "concrete benefits, not only professionally, but also in other areas of life", others might just hope to meet new people.
- *Subjective utility segmentation* takes both perceived costs and benefits into consideration. This approach is similar to expected value segmentation, but it construes a customer's preference as the difference between the weighted perceived benefits and the weighted perceived costs.
- The *role segmentation* classifies individuals based on their social positions, which presumably influence their choices. For instance it is reasonable to discriminate between employees, civil servants, and self-employed individuals because each of these groups prefers different products.

However, it can be difficult to obtain the relevant information from the customer, especially if it is not intuitively plausible why the situation demands that the information be revealed, as done in subjective utility and role segmentation.

In summary, attitude and expected utility segmentation appear to be the most promising psychodemographic techniques. Both were used to enhance the CBR sales agent described here.

7.2. Relevant Dimensions in the Travel Domain

In order to design an Internet travel agent, it is necessary to first identify the dimensions relevant for market segmentation in the travel domain.

7.2.1. Relevant Motives in the Travel Domain

Attitude segmentation can be applied to the customer's attitudes, as well as to her motives. An Attitude is defined as *a favorable or unfavorable evaluative reaction toward something or someone, exhibited in one's beliefs, feelings, or intended behavior* (Meyers, 1993). The term "motive" is a somewhat vague, as it is used differently by many authors.

E.g., Atkinson (1958) defines motive as a capacity, which is relatively stable in a given person but which differs from person to person, to gain gratification from a particular class of incentives.

The term "motive" will be used as interchangeable with "attitude" because of the fuzziness of the definitions. It is not important for our purposes to make a finer-grained distinction.

A lot of effort has been devoted to singling out the strongest motives for traveling. One of the largest German studies which have examined this issue is the "Reiseanalyse" study (Forschungsgemeinschaft Urlaub und Reisen e.V. (FUR), 1997). Its authors identified a set of motives based on qualitative longitudinal data collected from a representative sample.

According to this study, there are 27 different motives for traveling. Component analyses were performed to identify clusters of motives (Kanthak, 1973; Sommer, 1974; Lohmann & Wohlmann, 1987). This analysis yielded five groups of motives: recreation, freedom, adventure, social motives, and others. Table 7.1 displays the 14 most important submotives arranged in these groups.

7.2.2. Relevant Expected Benefits of Traveling

Expected benefits of traveling include all the perceived advantages of being at your travel destination vs. at home. When choosing between two destinations, those advantages will be weighted more strongly that are different for both places. E.g., spending a weekend in Paris gives plenty of opportunity for sightseeing but is not likely to help you relax.

The most important aspect in this context is probably the opportunity to engage in certain activities. Even though travel destinations differ with respect to a whole range of aspects, most of us will concentrate on those few activities that we've set our minds on. For instance, there are benefits that are intimately tied to a specific place or season (The opportunity to celebrate Carnival in Venice is of course tied to a certain city during a certain time in the year). Motives associated with such goals may strongly influence the customer's decision but they are too specific to be applied in a general travel agent.

The "Reiseanalyse" study also lists favorite vacation activities. People were asked: *This is a list of things you can do when you are on vacation. Which of these activities were part of your last vacation?* Table 7.2 shows the results of the "Reiseanalyse" study in 1989 (Lohmann & Besel, 1990). Giegler (1982) and Lohmann and Wohlmann (1987) also performed component analyses on leisure time activities and extracted components that are highly similar to those reported in the "Reiseanalyse" study. Thus, we can assume that those activities are sufficiently frequent and predictable.

7.2.3. Other Dimensions in the Travel Domain

Of course some additional information is necessary to aid customers in planning their trip, e.g., travel dates and the number of persons traveling. We will refer to these data as "hard facts", because although they may not be completely invariable, they represent product attributes. They thus represent a different type of information than abstract dimensions such as motives and activities. "Hard facts" are situated somewhere between customer interests and product attributes.

Customers expect to be asked for this information and can usually answer with high precision. Thus, these data should be collected before the customers are asked for more abstract information such as travel goals and interests.

TABLE 7.1.: Travelers' Motives. Adapted from Forschungsgemeinschaft Urlaub und Reisen (1997). People were asked: "The next topic is travels. There are some things that matter more than others when traveling. Please indicate which of the following aspects are most important to you when you go travel." Respondents could check more than one answer.

Motive	Frequency of agreement
Recreation motives	
relaxation	58%
recreation	46%
having fun	33%
Adventure motives	
seeing foreign things	30%
experiencing change	29%
experiencing foreign countries	29%
Freedom motives	
escaping from my daily routine	59%
being free to do whatever I want	38%
Social motives	
spending time with others (partner, friends, family)	40%
meeting new people	26%
having time to play with children	20%
Other motives	
experiencing nature	36%
caring for own health	29%
having fun	27%

TABLE 7.2.: Most popular vacation activities. Adapted from Lohmann et al. (1989). Includes only activities that were reported with a frequency of over 40.0%.

Activity	Frequency of agreement
Relaxation	
basking in the sun	59.3%
sleeping, relaxing	55.6%
lying on the beach	51.6%
Sport	
swimming	69.9%
hiking	45.9%
Hobby, fun	
writing letters	54.6%
taking pictures	53.0%
reading newspapers	51.7%
reading books	41.6%
Culture, education	
exploring surroundings	70.6%
sightseeing trips	55.2%
Active recreation	
walking	72.8%
Social activities	
talking to others	69.1%
meeting new people	41.8%
Other activities	
visiting restaurants, bars	67.5%
shopping	60.3%
eating specialties	46.0%

8. Implementing a Learning System

The previous chapters specified the dimensions underlying for customer behavior in general, as well as in the specific domain of vacation planning. We will use these predictors to design a user model of the customer. Further, A CBR-algorithm will enable the sales agent to select adequate products based on interaction with the customer. This chapter will describe these two components of the learning system in detail.

8.1. User Models

There are, in fact, two user models. The first one represents the customer's needs. Users are asked to explicitly state their priorities and expectations. Table 8.1 illustrates an exemplary user model of this first kind. This information represented in the first model is then used to design a second user model that specifies the product attributes. How this implicit inference is achieved will be described in a later chapter. See table 8.2 for an example of the second user model.

The unique characteristics of both models become evident when they are compaired with respect to the dimensions suggested by Rich (1983).

TABLE 8.1.: Example of a Type 1 user model (customer needs)

User model of needs and desires			
Motives		*Activities*	
Recreation motives:	very important	Relaxation:	yes
Adventure motives:	not important	Sport:	no
Freedom motives:	important	Hobby, Fun:	no
Social motives:	important	Culture, education:	yes
Other motives:	less important	Active recreation:	yes

Of course, both user models describe the individual customer, and not. The goal of modeling is not to identify certain types of users but to offer individual assistance to every user.

As mentioned before, the first user model is explicit and transparent. Values are defined directly by the user, who knows that this information will be used in a catalogue search. It is reasonable to assume that the user won't mind giving this rather generic information.. The second user model infers product attributes from the first model in a process that will be described in more detail later. In the beginning at least, the second model is implicit and not transparent to the user. However, if the user criticizes the initial offers made by the agent, the second user model becomes transparent so that it can be corrected and refined.

The refinement process brings to attention another difference between the two user models. While the first model is only used during the first retrieval and is therefore more or less static, the second model is dynamic. It is adjusted in an iterative process to achieve a high match with the goals of the user.

TABLE 8.2.: Example of a Type 2 user model (product attributes)

User model of attributes	
house ID:	76221F
region:	Bretagne
city:	Plozevet
price:	1116 DM per week
Description	
persons:	7
single beds:	4
double beds:	2
floors:	1
size of home:	80 m^2
premises:	3000 m^2
description:	beach 500 m
pets accepted:	true
Comfort	
bed for baby:	true
balcony:	false
TV:	false
yard:	true
fireplace:	false
parking:	true
Kitchen	
kitchen appliances:	false
dishwasher:	false
freezer:	false
microwave oven:	false
coffeemaker:	true
laundry machine:	true
Placement	
outside of town:	true
distance to downtown:	5 km
close to ocean:	true
close to beach:	true
close to lake:	false
close to river:	false
close to mountains:	false
close to tourist attraction:	false
close to restaurant:	false
Sports	
solarium:	false
whirlpool:	false
pool table:	false
bicycles:	false
tennis court:	false
table tennis:	false
boat:	false
swimming pool:	false
sauna:	false
dart board:	false

Both models can be characterized as long-term predictive models. User interests and product attributes are considered valid for the entire session, unless they are redefined.

8.2. Cases and Case bases

Two case bases administrate the user models. The customer case base contains cases of individual customers. The product case base provides cases of product attributes.

8.2.1. Customers Case base

All elements of in this case base share a common structure. Each case represents a set of needs (see appendix A.1 for a detailed exemplary tree structure). As shown in figure 8.1 the solution associated with each case consists of a complete set of product attributes. This task requires expert knowledge. However, only assumptions within

Case: Customer characteristics		
problem needs and desires	$\Rightarrow$	solution product attributes

FIGURE 8.1.: Construction of a case in the customer case base

the domain (here the domain of customer characteristics) are made. Only customer characteristics are determined. The relationship between customer characteristics and product attributes is learned and requires no prior assumptions, which distinguishes this system from others.

Local similarity measures. In order to convert a user model into a case, relevant local similarities have to be specified in addition to the given attributes. Local similarities describe the relation between the different kinds of attribute properties. Certainly these similarities depend strongly on the domain, but some principles can be generalized.

Expected gains are usually modeled as binary items. E.g., in the travel domain the customers are asked to state whether they are interested in *sightseeing and culture* or not. More detailed questioning is probably not useful and might confuse the customer. The similarity measure for binary items is defined as follows: Equal peculiarities are treated as maximally similar ($sim(query, case) = 1$), while unequal attributes result in a similarity of $sim(query, case) = 0$ as shown in table 8.3.

If no data are available because the customer refused to answer a question, then the corresponding attribute is treated as *undefined*. Undefined values may be interpreted as true or false, depending on the context.

TABLE 8.3.: Example of a similarity measure for binary attributes

		case	
		true	false
query	true	1	0
	false	0	1
	undefined	1	1

Motives and attitudes can be modeled with higher resolution. Psychological diagnostic questionnaires usually let the respondent choose among 5 to 10 categories. A similar resolution is recommended for data collected by a sales agent . Assuming that motives are interval-scaled, similarity measures can be conceived as assuming integer values (CBR98 (1998)):

$$sim(q,c) = 1 - \left(\frac{|q-c|}{ub-lb}\right) \qquad (8.1) \qquad \text{with} \quad \begin{array}{rl} \text{q} & \text{query} \\ \text{c} & \text{case} \\ \text{ub} & \text{upper bound} \\ \text{lb} & \text{lower bound} \end{array}$$

The greater the discrepancy between case and query the lower the similarity measure. Figure 8.2 shows different similarity values for a motive with a numerical ranging from 0 to 5. Again, *undefined* data are treated as true or false, depending on the context.

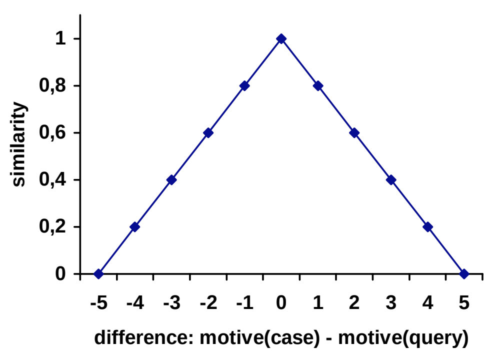

FIGURE 8.2.: Example of an integer similarity measure

Weights. The weight of each attribute depends strongly on the domain, too. In the travel domain, for instance, weights have not been used, such that all motives have generally been treated as equally important.

Even weights and similarities can be subject to learning. E.g., the customer may be given the opportunity to weight her preference for an attribute or a goal. This weight is used in the calculation of the overall similarity, The CBR algorithm offers methods for changing the weights as a function of experience, but for the simple prototype presented here it will be sufficient to assume a fixed value.

Computation of global similarity. When all local similarities and weights have been defined, the global similarity between two cases in the customer case base can be calculated (e.g., see Lenz, Auriol, and Manago (1998)). This calculation can be done in different ways. Options include using the minimum, the maximum, or the weighted mean of the local similarities. For our purposes we will use the weighted mean of all local similarities.

$$sim(Q,C) = \frac{\sum_{i=1}^{n} sim_i(q_i, c_i) \times w_i}{n} \qquad (8.2)$$

with

Q	query case
C	candidate case
n	number of attributes
q_i	i^{th} attribute of query
s_i	i^{th} attribute of case
w_i	weight of i^{th} attribute

The result is an ordered list of case solutions, sorted by the overall similarity with the query. Normally, only the best-fitting case is used for further processing. The result of the retrieval represents a translation of the customer's interests into concrete product attributes. In a second step of the retrieval that solution is treated as a query and matched with the elements stored in the case base of products.

8.2.2. The Product Case base

All elements of the product case base are based on the same model. Remember that this model is different from the model of customer cases. See appendix A.2 for a detailed tree structure of an exemplary case model for the travel domain. The solution associated with each case is, of course, a specific product (see figure 8.3). Thus, every product is represented by one element in the product case base.

Case: Products attributes		
problem product attributes	$\Rightarrow$	solution product

FIGURE 8.3.: Construction of a case in the product case base

Weights and similarities. The specification of weights and similarities follows the same principles that were applied to the customer case base. Several test trials for this project have shown that features should be weighted differentially. The price of a product, in particular, is usually a very important attribute and should therefore be weighted more strongly. Similarly, the travel destination weighs heavily into a customer's product choice.

Furthermore, some product attributes may demand for similarity measures different from those already discussed (e.g., see Bergmann (1998) for taxonomy similarity). The same is truefor case concepts. The concepts *vacation home* and *hotel accommodation* need to be discriminated, to name an example. A similarity measure for these two concepts does exist, since they share several attributes, yet each one of them also features attributes that do not meaningfully apply to the other one.

Price again offers a good example. Say you are dealing with the request "The price should not differ too much from the one that I specified" (see Equation 8.1). Rather than calculating a conventional similarity measure, it is appropriate in this case to follow the rule: "*less is perfect*". This involves the calculation of an asymmetric similarity that treats lower prices as more similar than higher prices. This is demonstrated in figure 8.4.

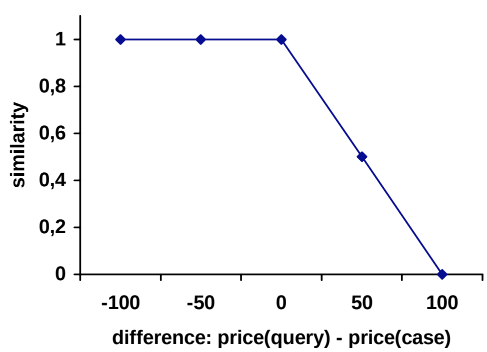

FIGURE 8.4.: Example of an asymmetric similarity measure for *Price*.

Note that modeling the product domain is not the primary objective of this paper. It will therefore not be described in more detail.

Computation of global similarity. Global similarity is computed in the same fashion as were global similarities in the customer case base. This product case base facilitates a comfortable attribute search because CBR eliminates the problem of near misses as well as the need for an "expert customer" (Wilke et al., 1998).

8.3. Learning Algorithm

So far we have assumed that the customer case base already exists. Let's consider how this case base is constructed in the first place.

As discussed earlier, the basic idea is to apply the system's experience to interactions with future customers. Consumers with similar needs probably choose similar products from the catalogue. Thus, if the system remembers interacting with a customer with the same interests as the current customer, then it should also remember which product was eventually chosen in the first case. If that product is no longer available, the system should be able to retrieve a similar one.

A new case is created every time a customer chooses a product from the catalogue. This customer is then treated as a single case—based on the pattern shown in figure 8.1. The user model specifying the interests of the customer constructs the case and the attributes of the chosen product are applied as the case solution. Weights and similarity measures are adopted from the basic case model. No further changes of the new case is necessary.

It would theoretically be sufficient to run the system after exposing it to a single case. For practical purposes, however, it is preferable to feed the system with a few representative cases before testing it. Once the system has accumulated enough experience, these pseudocases can be removed from the case base.

The learning algorithm is based on two assumptions:

1. A customer who chooses a product is satisfied with the attributes of that product. It represents the optimal approximation to the customers' needs.
2. The needs of a customer are stable. The suggestions made by the system have no impact on the customer's needs.

The first assumption is straightforward. Even if the customer's choice represents a compromise between what she wanted and what was available, she is probably satisfied with her choice. After all, there is no reason to purchase an unwanted product.

The second assumption may be considered more problematic. Imagine, for instance, a customer who is looking for a vacation home for her family. One of her favorite activities is swimming in the ocean. However, when browsing the catalogue she realizes that proximity to the ocean is related to price. Thus, she decides to choose a home further away from the beach that is much less expensive. Obviously, storing this choice as part of the new case is somewhat misleading, but since there is no real alternative we accept a violation of the second assumption.

The kind of learning we've discussed can be modeled with techniques other than CBR. E.g., neuronal networks seem to be quite effective dealing with the kinds of fuzzy associations we have considered. But CBR offers some unique advantages

Unlike neural networks, CBR allows for learning to be supervised and adjusted. It is possible, for example, to keep only those cases that were created after a certain date. This would guarantee that the case base is always up to date, allowing for new trends to be detected immediately.

It is also easy within CBR to eliminate cases that have become redundant. In some branches products change frequently. Consequently, cases linking to products that have been removed from the catalogue are no longer of use. Eliminating such cases is very easy in CBR systems.

Finally, even the case model itself can be adapted. If the representation of certain user goals or attributes turn out to be useless, perhaps because users refuse to answer the corresponding questions, then it is straightforward to adjust the case model accordingly without interrupting the retrieval process. In neural networks, on the other hand, all three scenarios can cause serious problems.

8.4. Example of an Interaction

This section presents an example of an interaction between a case-based sales agent (S) in the travel domain and a customer (C). During the interaction the user models are applied as shown in figure 8.5. The interaction is written in prose to make it easier for the reader to understand. It gives a feel for the type of interaction that is typical for virtual sales agents.

Step 1. First user model: In the first phase the system collects characteristics of the customers by asking them directly about their travel needs. Describing his goals and expected gains, the customer explicitly helps build the first user model.

S: *Where do you want to go?*
C: *I want to go to Brittany.*

S: *What are your favorite activities?*
C: *I prefer to have active holidays. Swimming and walking are my favorites. Relaxing is less important.*
S: *And what is the most important thing for you when you go on vacation?*
C: *I don't care much for culture or sightseeing. I like to meet people and to have fun.*

A simplified user model for this customer might look like this:

user model I	
region	Brittany
swimming	true
walking	true
culture	false
social motive	very important
adventure motive	very important

Step 2. First CBR retrieval: in a "fuzzy search" the system identifies the case in the customer case base that is most similar to the current model.

case

problem	
region	Brittany
swimming	false
walking	true
culture	false
social motive	very important
adventure motive	very important

⇒

solution	
region	Brittany
city:	Plozevet
price:	1116 DM per week
persons:	7
single beds:	4
double beds:	2
floors:	1
size of home:	80 m^2
premises:	3000 m^2
⋮	

Step 3. Second user model: The result of the first retrieval is a complete set of attributes of the product. These data are used to build the second user model.

user model II	
region	Brittany
city:	Plozevet
price:	1116 DM per week
persons:	7
single beds:	4
double beds:	2
floors:	1
size of home:	80 m^2
premises:	3000 m^2
⋮	

Step 4. Second CBR-retrieval: The system searches the product case base for a case with attributes that optimally match the query specified by the user model.

Step 5. Presentation: The result of this retrieval is a list of products that optimally meet the criteria at hand. The list is presented to the customer.

S: *These are the five apartments from our catalogue that best fit your description:* ... (a brief overview is presented)

C: *Oh, the second one sounds interesting. Can you describe it in more detail?*

S: ...

Step 6. Refinement: The customer may now impose further restrictions, thus explicitly refining the second user model. If this is the case the interaction proceeds to Step 4.

C: *I like this apartment. But it is a bit expensive. Do you have cheaper options?* Hence, the user model is refined by reducing the price:
e.g., *new price* = *old price* $\times$ 0.8

S: *These are the five apartments from our catalogue that best match your description:*
⋮

Step 7. Decision: The customer chooses a product. The interaction comes to a close.

C: *This option is very attractive. I'd like to commit to it.*

S: ...

Finally, the system is able to add a new case to the customer case base, representing the initial user model as the "problem" and the attributes of the chosen product as the "solution".

The more experienced the system, the better its performance. After it has dealt with a reasonable number of cases, the system is able to react flexibly to customer requests.

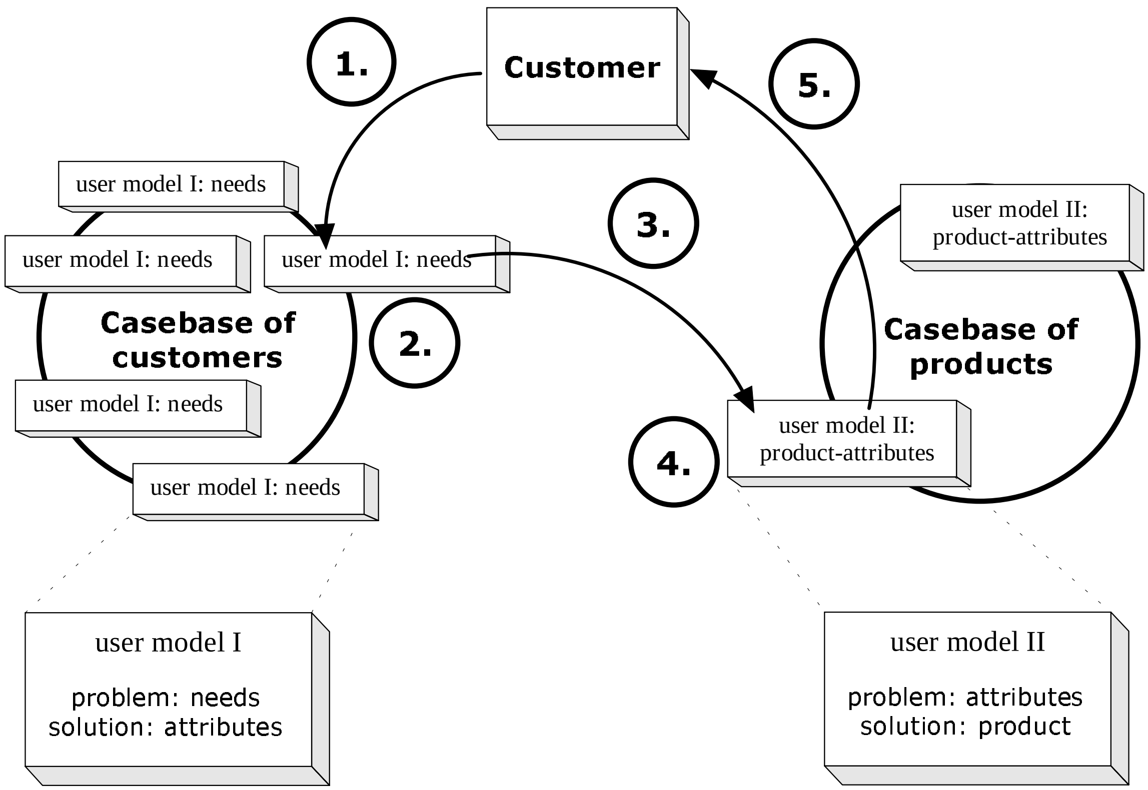

FIGURE 8.5.: Information processing with two case bases

9. Description of CASTLE

We will now examine an application of the general principles discussed so far: an electronic travel agent named CASTLE that helps customers find a vacation home. Since its design closely follows the principles described earlier, only those features will be described in detail that are unique to CASTLE.

9.1. The CBR-Engine

CASTLE is based on CBR-Works[1]. This software package is a comprehensive tool for the creation of searchable case bases. It also offers several interfaces that allow the sales agent to send requests to the case base.

CASTLE is composed of various CGI[2]-scripts that manage user sessions, send request to the CBR Engine, and present results. Figure 9.1 presents an overview of the CASTLE architecture.

In the first phase (henceforth referred to as "psychological component") of its interaction with a customer the system collects data about the customer. For each customer, three HTML forms are created so that once information has been entered it can be retrieved at any time by returning to the corresponding page. The data are sent to the CBR engine to compute the similarities between the new customer's profile and the cases stored in the case base. These steps are performed by code written in a programming language called Case Query Language (CQL). The resulting attributes are used for a second retrieval, this time from the vacation home case base.

The result of this retrieval, i.e. the solution, is presented to the customer. This is done by filling in slots on several templates. Which template is chosen depends on the amount of detail required. After this, the customer has a chance to start a refinement process, or else to choose a product.

[1]CBR-Works is a case-based reasoning product family from tecInno, Germany.
[2]Common Gateway Interface, used for interaction between user and server.

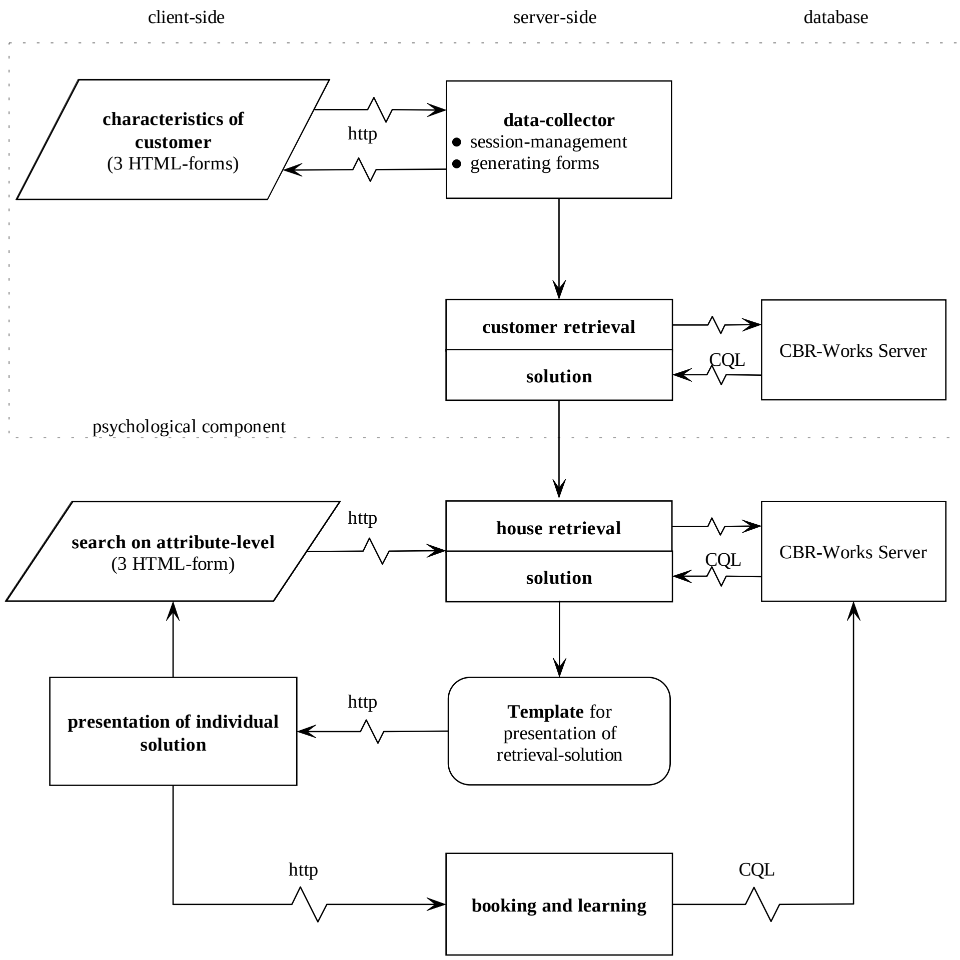

FIGURE 9.1.: Architecture of CASTLE

9.2. Design

Data about the user's motivation for traveling and his preferred vacation activities are collected with questionnaires that the user is asked to filled out. Table 9.1 shows an example. The customer can rate the importance of a certain motivation on a five-point scale. Vacation activities are described verbally and in pictures; response options are *yes, no*, and *undefined*.

A similar methodology is used to have the user refine the attribute user model, which will be translated into a product case. E.g., customers are asked for yes-no answers, which are then treated as values of Boolean attributes (e.g., *should have a deepfreezer*), or, in other cases, choose between more than two categories treated as attributes of type integer. An example of the latter is *price*: *about 500 DM, about 750 DM, about 1000*

TABLE 9.1.: Example of the input of motives with CASTLE

It's important for me ...

	doesn't matter	not important	important	very important	absolutely
... to relax	○	○	○	○	○
... to recreate	○	○	○	○	○
... to have fun	○	○	○	○	○
⋮	⋮	⋮	⋮	⋮	⋮

DM. All attributes are grouped thematically, as indicated in table 8.2, which simplifies further processing.

9.3. Interacting with CASTLE

CASTLE is intuitively designed and, hence, very user-friendly. After viewing a welcome page and some general information about the system, customers are asked to describe what, in their opinion, defines a perfect vacation. They state where they want to go and how many persons are traveling with them. Then they are prompted to list desired activities and other priorities.

Based on this information CASTLE retrieves a first offer. A list of five vacation homes is presented, with general information (e.g., *number of bedrooms*) describing each home, as well as a picture. The customer may inquire about further details, or begin a new search after changing some of the information. This process is repeated until a choice is made, or until the customer quits CASTLE.

Part IV.

Evaluation

A case-based user modeling system has been proposed that holds the promise of improving conventional electronic sales catalogues. An important dimension on which to evaluate this system is that of customer satisfaction. The following chapters report two experiments that were conducted to obtain empirical data on this subject.

10. Study Design

Evaluating a software system such as the sales agent considered here poses various challenges, two of which are questions about the validity and transferability of the results to other systems. It is important to recognize that the experimental subjects in this case are customers and therefore enter the experiment with a set of beliefs, expectations, and goals. This will be discussed further in section 10.2.

As is often the case in psychological experiments, there is a potential tradeoff between ecological validity and the controllability of the experimental situation. In awareness of this problem, the study was conducted in two phases.

The first experiment, the "Internet test", was implemented under ecologically valid conditions. It involved the observation of real customers, whose transactions were logged on the computer under two distinct conditions. The second phase was an experiment conducted under laboratory conditions with an between subject design. This strategy makes it possible to compare customer satisfaction across the two situations and to test two hypotheses:

Experiment I: A case-based sales agent learns from its experience interacting with customers. Its ability to respond to invidual queries improves gradually. Thus, Customer satisfaction is predicted to be a function of the system's expertise.

Experiment II: As measured in customer satisfaction, the case-based sales agent is superior to an electronic agent that does not employ user modeling, and to conventional electronic catalogues.

10.1. Experiment I: Internet Test

In the first phase of the study the system builds the customer case base by interacting with real customers. People interested in renting a vacation home in France were able to use CASTLE from their home computers while logged in to the Internet. The customer case base was initialized with 14 artificial cases judged to be realistic. Note that this was not necessary, as it is theoretically sufficient to feed the model with a single case that does not even need to be realistic. But in order to accelerate the learning phase expert knowledge was used.

10.1.1. Method

CASTLE was announced at several popular search engines of the Internet. Over a period of 6 weeks, 38 individuals visited the Internet sales agent to rent a vacation home. These customers are treated as group 1 (G_1). After this period, the case base was updated with the new information. 10 further customers, the second group (G_2) were observed during the following 3 weeks. All users that ended up not purchasing a product were excluded from the analysis. No exact data are available about this drop-out, however, it is estimated at about 90%.

10.1.2. Criteria

Customer satisfaction is a complex variable that is difficult to measure. In this context it was defined as follows: Customers are more satisfied with an interaction if (a) they don't have to go through long lists of attributes; (b) if they get an opportunity to express what they want in terms of goals and interests, rather than in terms of product attributes; (c) if a choice can be made after a relatively short interaction. No direct measures were used.

Most customers do not want to spend more time than necessary interacting with the agent. Our agent's "psychological component" is expected to speed up the interaction in a way that enhances customer satisfaction. Thus interaction time in the different phases can be used as criterion for customer satisfaction.

Different measures were used for defining consumer satisfaction:

- Total interaction time (T), interaction time before the first retrieval (T_1), and interaction time between the first retrieval and the purchase (T_2).
  Note that $T = T_1 + T_2$.
- Total number of aanswers given by the user (A), answers before the first retrieval (A_1), and answers given after the first retrieval (A_2). $A = A_1 + A_2$.
- Number of retrievals after the first retrieval (R).

Start	Search on the level of customers' characteristics • duration (T_1) • number of attributes (A_1)	First retrieval	Search on the level of attributes • duration (T_2) • number of attributes (A_2) • number of retrievals (R_2)	Booking

FIGURE 10.1.: Phases of a session and related measures

Figure 10.1 gives an overview which measure was applied in which phase of interaction.

10.1.3. Statistical Hypotheses

Two statistical hypotheses can be derived from the conceptual considerations described above. Participants using the inexperienced system (Group 1, G_1) should be less satisfied than those who use the experienced system (Group 2, G_2):

1. T is less for G_2 than for G_1. While T_1 is not expected to change, T_2 should be considerably shorter for G_2. T_1 is used as a baseline measure that allows to control for variations in extrinsic variables such as *individual differences in response time*, or *slow Internet connection*.

$$\frac{T_2(G_1)}{T_1(G_1)} < \frac{T_2(G_2)}{T_1(G_2)} \tag{10.1}$$

2. The amount of information provided by the customer (A) and the number of retrievals (R) are less for G_2 than for G_1.

$$A(G_1) > A(G_2) \tag{10.2}$$
$$A_2(G_1) > A_2(G_2) \tag{10.3}$$
$$R(G_1) > R(G_2) \tag{10.4}$$

10.2. Experiment II: Laboratory Test

The second phase of the study was conducted in the laboratory. Voluntary Participants were recruited at the campus. A small amount of money was provided for completing the experiment.

Unlike Experiment I, Experiment II of the study was not ecologically valid in several respects. Participants were aware of the purpose of the experiment; they participated in a virtual, as opposed to a real, sales transaction at the end of which they were forced to choose a product. However, there is no reason to assume that the relationship between customer satisfaction and other variables should be affected by the artificiality of the situation.

From a statistical point of view a within subject design is preferable, but thematically spoken consumers as participants cause some problems in this context. Once a user found the perfect home, she cannot be forced to perform another search under another condition, since she is already familiar with the content of the catalogue.

10.2.1. Method

Participants were invited to take part in an experiment that involved *Vacation homes in Brittany* and *Surfing the Internet*. At the beginning of the experimental session participants received the following instructions:

> Welcome to CASTLE!
> You will be using a newly developed vacation home catalogue.
>
> - CASTLE is an online-travel-agency.
> - CASTLE offers you a variety of vacation homes in France.
>
> *Please try to imagine the following situation:*
> You want to spend your next summer holiday in France. You decide to look for suitable accomodation. Searching the Internet you find an electronic catalogue.
>
> *And this is your task:*
> Please search this catalogue for a vacation home that suits your needs. We ask you to continue searching until you have found a home that meets your personal criteria. It should be acceptable in terms of its price, features, comfort, etc. You should be able to imagine spending your holidays there.

A minimum session length of 15 minutes was imposed to prevent participants from ending their search prematurely. Experiment I of the study had shown that it takes about 15 minutes to familiarize oneself with the system. None of the participants in Experiment II chose a house before the end of this period.

A total number of 62 subjects were observed.

10.2.2. Criteria

To test the hypotheses, participants were assigned to one of three conditions:

Experimental group 1 (G_1): In this condition participants interacted with the same version of CASTLE that was used in Experiment I.

Experimental group 2 (G_2): This group used a system without user modeling. Thus, instead of first exploring customer characteristics, the system started with an attribute search (see figure 10.1).

Control group (G_3): The third group used a simple electronic catalogue organized by geographical region. No other search criteria were available. This electronic catalogue contains almost the same information as a hard-copy catalogue available at travel agencies. The order in which products are listed within a category is random.

The dependent variables were the same as in Experiment I but one additional variable was assessed. In a similar evaluation Strachan, Anderson, Sneesby, and Evans (1997) administered a questionaire measuring user satisfaction for comparing an adapting system to an usual system. The *questionnaire of User Interaction Satisfaction (QUIS)*, developed and revised by Chin, Diehl, and Norman (1988), is commonly administered in HCI research to compare different types of interfaces with regard to user satisfaction. QUIS has been shown to yield highly reliable data (Harper & Norman, 1993; Slaughter, Harper, & Norman, 1994).

A set of 17 items from the QUIS main scales that were believed to be useful for the present purposes were translated into German. Items that could not be applied in this context were not used. E.g., *Remembering names and use of commands was difficult* was not a useful item since there are no commands in CASTLE.

Two items were added. The first one explicitly addresses the degree to which the chosen product meets the customer's expectations: *The vacation home that I chose has what I was looking for.*
The second item addresses the user's certainty that the optimal product was found: *I think there is a more suitable vacation home in the catalogue.* Both items should also measure customer satisfaction but are more closely related to the experimental task than the QUIS items. See appendix B for a listing of all items.

Another possibility of measuring customer satisfaction would be to develop a questionnaire assessing the perceived quality of service (Hurley & Estelami, 1998). Such a tool would assess the customer's overall satisfaction and thus perhaps be more attractive for companies. Unfortunately, the development of such a questionnaire is beyond the scope of this project.

10.2.3. Statistical Hypotheses

Again, two statistical hypotheses were tested:

1. Participants in G_1 should on average be more satisfied with the interaction than participants in the other two groups:

$$T(G_1) < T(G_2) < T(G_3) \tag{10.5}$$

$$A(G_1) < A(G_2) \tag{10.6}$$

$$R(G_1) < R(G_2) \tag{10.7}$$

2. Customer satisfaction as assessed by the QUIS should also differ between the groups:

$$QUIS(G_1) > QUIS(G_2) > QUIS(G_2) \tag{10.8}$$

11. Results

Results are reported in three sections. Section 11.1 describes the cases from Experiment I that were used in Experiment II. Sections 11.2 and 11.3 discuss the outcomes of the study with regard to customer satisfaction.

11.1. Description of the Learned Cases

The cases in the customer case base can be described in several ways. In order for any case base to be useful, there must be variability among its attributes, as features that do not vary between cases are useless. Table 11.1 shows that motivational criteria vary considerably between cases. Only *recreation* lacks variability ($SD = 0.22$) and therefore adds little to the prediction. This criterion can thus be omitted from future versions of CASTLE.

Another way to describe a case base was introduced by Smyth and McKenna (1998). According to these authors the competence of a case base is measured in terms of case base size, case density, and case base coverage.

Size is indeed a plausible, but not a sufficient measure, since even small case bases can perform well. A total number of 49 new learned cases is certainly not too much but since effects can be observed even with this case base size for this purpose it seems to be sufficient. In addition, this is not a problem arising from the architecture of the system. The case base will grow considerably fast since a new case is created with almost every new customer. Only doubled cases are omitted.

The case density (the authors term is somewhat misleading here) of a single case c in a group G within a case base C with $G \subseteq C$ is calculated with the following equation:

$$CaseDensity(c, C) = \frac{\sum\limits_{c' \in G - \{c\}} sim(c, c')}{|G| - 1} \tag{11.1}$$

The group density is the average case density of the group:

$$GroupDensity(G) = \frac{\sum\limits_{c \in G} CaseDensity(c, G)}{|G|} \tag{11.2}$$

Finally coverage of a group is calculated as follows:

$$GroupCoverage(G) = 1 + [\,|G| \cdot (1 - GroupDensity(G))\,] \tag{11.3}$$

This measure allows to compare two groups of cases within a case base. Of particular interest is the comparison between the artificial cases with which the case base was initialized, and the "real" cases subsequently stored in the case base. This comparison can serve as an estimation for the quality of the case base.

Three subgroups of cases from the final case base are compared. The first group (C_1) is a hypothetical group. It contains only one single reference case, as a single pseudocase

TABLE 11.1.: Agreement to motives and corresponding variability

Motive	$\overline{\mathbf{x}}$	σ^2
Recreation motives		
relaxation	0.47	0.60
recreation	0.95	0.05
having fun	1.85	0.63
Adventure motives		
experiencing change	1.38	0.97
experiencing foreign countries	0.53	0.49
Freedom motives		
escaping from my daily routine	2.30	0.40
being free to do whatever I want to	1.71	0.76
Social motives		
spending time with others (partner, friends, family)	0.74	0.20
meeting new people	1.36	0.55
having time to play with children	0.68	0.56
Other motives		
experiencing nature	1.91	0.52
caring for own health	0.41	0.25
having fun	0.81	0.62

could have been used in the begining. This group provides a baseline measure. The second group (C_2) consists of the 14 artificial cases that represented the case base for G_1 in Experiment I. Finally, the third group of cases (C_3) contains all available cases, i.e., the 14 artificial cases plus all "real" cases.

TABLE 11.2.: Results of the comparison between different case bases

	C_1	C_2	C_3
case base size	1	14	54
group density	1	.552	.497
group coverage	1	7.272	28.14

The results of this comparison are shown in table 11.2. Since there is no similarity between cases in C_1, group density is 0 by definition. In the other groups, density differs slightly around .5, with the density of C_2 being somewhat greater than the density of C_3. Group coverage varies considerably between groups. According to Smyth and McKenna (1998), this should be interpreted as indicating that C_3 has the highest competence in solving problems. Thus, the learned cases not only extended the case base but also enhanced the competence of the system.

11.2. Results of the Internet Test

Users interacted with the system for an average of 555 seconds , initiating an average of 1.58 retrievals.

Table 11.3 displays the main results of the comparison between the groups. Due to the

TABLE 11.3.: Overview of the results of experiment I

criterion	group 1	group 2	T	df	α	power[a]
total duration T (s)	628	436	1.79	19	.089	.28
duration of search T_2 (s)	390	274	1.24	19	.230	.28
proportion $\frac{T}{T_2}$	1.80	1.85	-0.10	19	.922	.28
decisions A	82.4	65.1	0.92	19	.369	.28
refinements A_2	56.8	39.0	1.04	19	.313	.28
retrievals R	3.08	1.87	1.16	19	.262	.28

[a] $power = 1 - \beta$ with $\alpha = .05$ and an assumed effect-size of .5

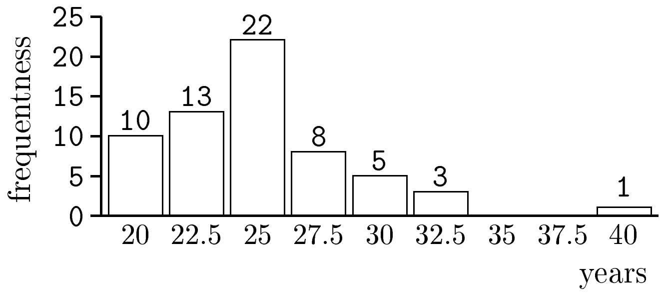

FIGURE 11.1.: Age

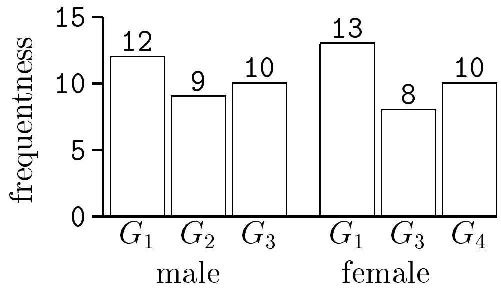

FIGURE 11.2.: Gender

arrangement of questions (several questions on one page) the exact number of decisions (A) and the number of refinements (A_2) cannot be determined. We assume that every time a page is presented the user answers all questions. So the total number of decisions (A) is defined by the questions per page (n) and the number of presentations of this page (P):

$$A = \sum P_i \times n_i \tag{11.4}$$

People probably spend less thought on single attributes when presented repeatedly with the same list of questions but they still answer each question. Thus, equation 11.4 is a fair estimation of the number of decisions.

As predicted in hypotheses 10.2 - 10.4 group 2 shows increased satisfaction. Due to the small sample (N=48) the differences did not reach statistical significance. Nevertheless, it is remarkable that almost all results point in the same direction, with the exception of the proportion $\frac{T}{T_2}$ (hypothesis 10.1). The groups do not differ on this measure.

Power calculations show that there was only a small chance of finding a medium-size effect in all analyses. Due to the nature of the study, it was difficult to achieve a larger sample size. Finding real customers on the Internet usually requires a great deal of advertising, which was unachievable in this study. Given these technical limitations, the results from Experiment I are very encouraging.

11.3. Results of the Laboratory Test

The distribution male and female participants is equal across groups (figure 11.2). Due to the mode of recruitment of subjects the age range is not as wide-spread as desirable (figure 11.1). The average age is 25.1 years.

Subjects were asked to rate their experience with computers and the Internet. Table 11.4 shows the distributions of these ratings. These data were collected because participants in previous studies (e.g., Strachan et al. (1997)) had surprisingly little computer experience, which introduced a confound in the participants' satisfaction with the system. In the present sample, however, participants were sufficiently familiar with using computer and the Internet.

A participants' satisfaction score on the QUIS is given by the participant's mean response. The direction of item 16 had to be changed before computing the mean. High QUIS scores indicated high satisfaction.

Item analyses showed excellent reliability: the Guttman split half reliability of the original QUIS items was .805. The 2 non-QUIS items that were added to the questionnaire were only moderately correlated (Item 16: r=-.361, Item 17: r=.305) with the original QUIS items. Moreover, the correlation between these two items was surprisingly low (r=.094). Since the items correlate with the QUIS, a possible explanation for this result is that the two new items measure different aspects of satisfaction.

A one factor ANOVA was performed for every criterion to test the statistical hypotheses. The results are shown in table 11.5. As in Experiment I, there is a trend of all measures in the predicted direction, although only two results reach statistical significance. The duration of search (T_2) was clearly smaller in G_1. Planned comparisons between the groups showed a significant difference ($t(40) = -10.72$ and $t(43) = -13.26$).

This result can be interpreted in two ways. It is possible to attribute the difference in consumer satisfaction measures to the performance of the system. Participants in G_1 were probably presented with better first offers than participants in G_2. However, the effect may also represent an artifact. It is conceivable that participants used the *15 minute rule* as an heuristic for when to end the session. Due to the nature of the experimental design, T_2 was shorter in G_1 when the interaction was finished after 15 minutes, because this group first had to answer questions about their goals and interests.

TABLE 11.4.: Self-rating computers and Internet experience

How often do you use...

	...computers	**...the internet**
never	1	4
seldom	4	8
casually	13	22
several times a week	26	20
daily	18	8
$\overline{x}$	2.90	2.32
σ^2	0.91	1.14

TABLE 11.5.: Overview of the results of experiment II

criterion	G_1	G_2	G_3	F	df	α	power[a]	ω^2
total duration T (s)	1057	1082	1113	0.25	2, 59	.782	.39	-
duration of search T_2 (s)	352	1082	1113	81.2	2, 59	.000	-	.72
QUIS	5.75	5.29	6.27	1.93	2, 59	.155	.39	-
decisions A	123	147	-	2.56	1, 39	.568	.35	-
refinements A_2	95	147	-	102	1, 40	.027	-	.70
retrievals R	7.48	8.05	-	0.16	1, 40	.688	.35	-

[a] $power = 1 - \beta$ with $\alpha = .05$ and an assumed effect-size of .25

Thus, T_2 may not reflect the actual duration of search, but no better approximation is available.

The number of refinements (A_2) also differs significantly between the groups, with G_1 making less refinements on average than G_2. Again, it may be suspected that this is an artifactual result. However, participants in G_1 had fewer retrievals and were more satisfied.

Because none of the main effects were not statistically significant, no further analyses were performed on these data. As mentioned earlier, a within-subject design would have been preferable from a statistical point of view, but practically impossible. In order to reduce error variance, a multiple regression analysis was performed, using *experience with the Internet* and *gender* as additional predictors. The reason for this was that ANOVAs (the results are not reported here) had shown significant effects of those two variables. The results of the multiple regression model with the crite *total duration, QUIS*, and *number of refinements* is displayed in table 11.6. A summary of results of the multiple regression analyses is shown in table 11.7.

Using the variables *experience* and *gender* as predictors improves the quality of prediction only for T. Not surprisingly, the more experience participants had, the faster they interacted with the system. Males found a home that suits their needs faster than females. The other criteria are not explained better when the additional variables are entered into the regression.

TABLE 11.6.: Summary of the multiple regression for the variables *total duration (T)*, $QUIS$, and *number of refinements* (A_2)

criterion	predictor	B	std.error	Beta	T	sig.
duration (T)	constant	1008	193.1	-	5.22	.000
	group	34.0	90.6	.06	0.38	.709
	experience	21.9	41.6	.10	0.53	.602
	sex	-70.1	87.6	-.13	-0.80	.428
QUIS	constant	5.57	0.99	-	5.63	.000
	group	-0.36	0.47	-.13	-0.77	.448
	experience	0.14	0.21	.12	0.69	.494
	sex	0.35	0.45	.13	0.78	.441
refinements (A_2)	constant	39.2	51.4	-	0.76	.451
	group	53.8	24.1	.35	2.23	.032
	experience	3,99	11.1	.06	0.36	.721
	sex	-14.1	23.3	-.09	-0.60	.549

TABLE 11.7.: Summary of the significance of the models shown in table 11.6

criterion	R	R^2	df	F	sig.
T	.178	.032	3, 38	0.412	.745
$QUIS$	.221	.049	3, 38	0.649	.588
A_2	.361	.130	3, 38	1.899	.146

12. Conclusion

The results of both experiments confirmed the predictions. Although statistical significance was not always reached, clear trends were observed. In most cases, the lack of significance is due to low statistical power.

The first hypothesis stated that a case-based sales agent improves as function of experience. It was shown that even little additional experience is capable of changing the way the system works. In a commercial system, thousands of requests per day are not unusual. Under such conditions, a very elaborated case base is built up very quickly, and the system could be further improved.

The present findings strongly encourage the use of learning systems. Instead of consulting experts on various aspects of the user, these systems actually collect relevant data from the user. There are still a lot of assumptions in a sales agent, so we should make use of Designers of Internet sales agents. These systems do not let the opportunity pass to receive this kind of empirical information.

The second hypothesis was that the *psychological component* of systems such as CASTLE would improve customer satisfaction. Two results are interesting with regard to this hypothesis.

First, case-based sales agents seem to be more popular than choose-out-of-a-list systems. Most existing systems can thus be improved. The *fuzzy search* seems to be closer to what customers expect from a sales agent. The quality of the offers further improves when a CBR algorithm considers not only the wording of the query but also the structure of the domain.

Secondly, user modeling is superior to the traditional CBR approach, i.e. the two-step system with two user models worked better than the one-step system. Of course, user modeling is not applicable in every situation. But especially in complex domains with product categories that contain numerous structurally similar elements, pre-selection based on customer characteristics appears to be useful.

Customers seem to appreciate being given the opportunity to describe their goals and interests. Only few people refused to answer these questions; most customers were more satisfied when they had this opportunity.

In summary, this project provides strong evidence that case-based learning systems can improve sales support considerably.

Part V.

Future Perspectives

13. Further Improvement of the Implementation

A lot of work was done on the implementation of CASTLE but of course this system is still a prototype. There are still many possibilities of improvements which might be considered in future implementations of case-based sales agents. Some of them will be described in this chapter.

13.1. User Models

In the current implementation customer user models were only used once. A new approach could try to draw more information from the user models.

For instance, during the query collection phase in the beginning, a sales agent could profit from the user model in the following way: After every question the current user model is compared to the cases in the casebase of customers. Which question the system will ask next depends on the expected information gain. That is, the increment of similarity for every attribute is calculated *on the fly*. If a further question would help decide between two customer cases this question will be selected. A question that can't provide additional information because the remaining cases do not differ on this attribute can be skipped. This procedure minimizes the amount of questions necessary for identifying the matching case. The same procedure is certainly applicable during the second step with the user model of attributes.

In the presentation phase a sales agent could profit from the user models, as well. By comparing the product attributes and the user model at hand the important information could be highlighted. E.g., all those attributes that are in conformity with the query attributes should be emphasized. There could also be a special hint to attributes that differ extremely from the query. This would support the customer in getting a quick impression of the complete product.

Another possibility of strengthening the utilization of user models is to extend the model by a temporal component. By logging the interaction moves of the customer the sales agent could learn to react perfectly. The system could anticipate certain steps that did occur frequently with former customers. An exemplary association might be that customers who change the attribute *number of persons* frequently also change the attribute *living space* in the same direction. In this case the system could automatically infer this second refinement and present corresponding offers.

13.2. Interaction

Besides changing the user model the interaction itself could become more dynamical and natural.

According to some informal feedback many customers want to have the opportunity of defining the importance of an attribute. Similar to the statement *there should be a swimming pool*, as it is typical for CASTLE, customers should be able to express statements such as *there must be a swimming pool* or *I prefer a swimming pool but I don't mind if there is no*. In CBR language this means the customers can change the weights of attributes and set filters. On the one hand this is easy to implement in a CBR system since weights (though set to a fixed value here), are part of the algorithm anyway. On the other hand the number of features and possibilities should not overload the users, which means, variability of inputs should not degenerate. An implementation has to negotiate these two points.

In addition, a dialog planner might improve the interaction. This kind of subsystem is used in other systems such as VIPER in PRACMA (Weis, 1994). Such a planner could select dialog moves depending on meta-plans and strategies. Unfortunately the internet sets some limitations here since a real-time interaction is impossible.

13.3. Learning

Finally the complete algorithm of learning can be improved. As mentioned before, the system could learn weights and even similarities.

If the customers also rate their preferences in terms of importance, as proposed above, these weights might become part of the user models and finally part of the learned cases. When referring to this case the weights can be used for the further computation of the global similarity as long as the current user did not define new weights.

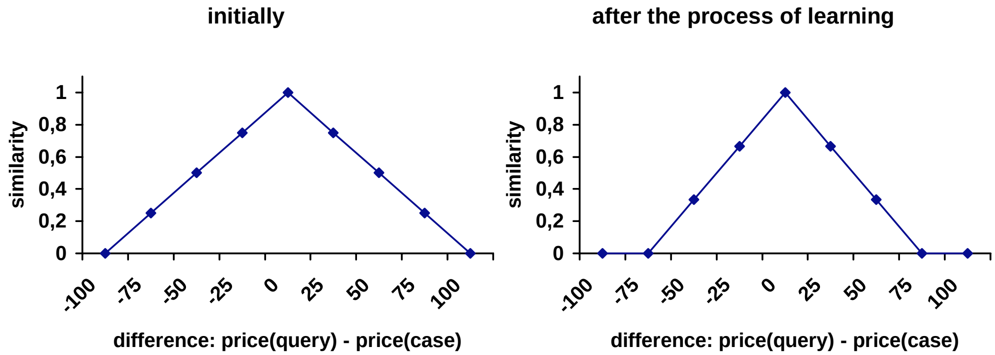

FIGURE 13.1.: Exemplary adaptation of the similarity of price before and after the process of learning

To learn similarity measures is a somewhat more complex task. Especially if individual similarities are to be inferred, it is difficult to get the required information. Which cases are perceived as similar by the user? Not to mention the quantitative value of similarity. And which attribute of the product is responsible for this difference of the global similarity? In such a short duration of interaction—compared to other software systems that might collect data about the customer over a period of days or even months—a valid inference is almost impossible.

This is not true for overall similarity measures. Assuming there is only one similarity measure per attribute for all users a step by step adaptation of these measures might take place depending on the experience of the system. Two examples will illustrate this procedure in integer and binary variables. The sales agent might find customers frequently rejecting products that differ extremely in price, while the other attributes are pretty close. An adaptation of the local similarity of price would help in future interactions. In this example the similarity of extreme differences in price should be reduced. Figure 13.1 displays an exemplary adaptation of similarity measure of price.

But also the opposite might happen. People might frequently accept products that do not have a feature that was defined as preferred before. The nonexistence of this attribute might be treated as more similar in an experienced system. This kind of stepwise adaptation of similarity measures would reduce the amount of necessary inherent assumptions about the domain.

Besides this detail of inference there might be other methods of finding the best match between customers preferences and product attributes. Neural networks are famous for learning unexpected nonsense associations. But, as argued before, in this domain CBR has some advantages that plead for CBR: The maintenance of the data is easier and learning can be supervised and influenced when using a (symbolic) casebase. Finally in

CBR a theoretically or empirically based approach can be applied for example by fixing certain weights.

After all, we can question the system in depth. A two-step system did work pretty well in this case. But is this procedure really necessary? An important reason for this way of conceptualization was the plausibility. But maybe there are other approaches that show even better results. It would be interesting to find out evidence about the thesis that a two step system reasons closer to the way a real salesman does than other systems achieve.

14. Further improvement of the Evaluation

Finding the optimal way of evaluation of a complex software system is very difficult. At least three additional aspects might be considered in future evaluations.

On an elementary level an evaluation could proof some inherent assumptions about how the system works. Instead of testing the overall quality of the system, which is influenced by many uncontrollable factors, a more detailed evaluation could be applied. Besides the test whether the system is really learning—as done in this work—a similarity test might be established: Do users rate objects as similar that are similar according to the system? Are users more satisfied when they get the perfect offer at once? Would experts select the same product based on the given information? This kind of evaluation would help to find different inherent errors that would otherwise remain hidden by the variability.

The evaluation described in this work is designed to encourage further work on case-based reasoning by comparing this inference technique to standard systems. But an other comparison would also be of interest. Does CBR work better than other inference methods such as Bayesian Networks or discrimination networks. Where are the advantages and disadvantages of inferring information by these techniques. With the abstract criterion *customer satisfaction* a direct contest might be established since this should be the main goal of all systems.

Another subject of evaluation could be the *intelligence* of the sales agent. Does a sales agent really act like a salesman in a shop? An experiment that is designed in the following way might answer this question. Turing (1950) proposed a concept for the decision whether a machine can think. This test—now called the Turing Test—uses two interfaces (those days two teletypewriters). One of the interfaces is connected to the tested system the other one is connected to a human. If a person interacting with both "systems" cannot tell which interface is connected to the human, then the computer is "thinking". A future evaluation could compare the reactions of a sales agent to the reactions of experts in the domain.

This kind of test would point out the competence of a system but is this the optimum? Maybe some day nobody will dare to go to a shop without his or her personal sales agent. Or maybe some day nobody dares to interact with sales agents since these creatures became too intelligent succeeding in talking us into buying products that we don't need at all.

Appendix

A. Tree-structure of Case bases

A.1. Case base describing Customers

This is the model of a case describing a customer. *Concepts* are written in Italics. Attributes are indicated with the associated type in brackets.

Customer

- *Preferences for activities*
  - relaxation (Boolean)
  - active recreation (Boolean)
  - culture, education (Boolean)
  - hobbies, fun (Boolean)
  - shopping (Boolean)
  - social activities (Boolean)
- *Recreation motives*
  - relaxation (motive type)
  - recreation (motive type)
  - enjoying myself (motive type)
- *Adventure motives*
  - seeing foreign things (motive type)
  - experiencing change (motive type)
  - experiencing foreign countries (motive type)
- *Motives of freedom*
  - escaping from my daily routine (motive type)
  - being free to do whatever I want (motive type)
- *Social motives*
  - spending time with others (partner, friends, family) (motive type)
  - meeting new people (motive type)
  - having time to play with children (motive type)
- *Other motives*
  - experiencing nature (motive type)
  - caring for own health (motive type)
  - having fun (motive type)
- date of case construction (for maintenance purposes only)

A.2. Case base of Products

This is the model of a case describing a product. *Concepts* are written in Italics. Attributes are indicated with the associated type in brackets.

Vacation home

- home ID (String)
- *place*
    - region (region type)
    - city (String)
- price (price type)
- *description*
    - number of persons (person type)
    - number of single beds (integer)
    - number of double beds (integer)
    - number of floors (integer)
    - size of home (integer)
    - size of premises (integer)
    - text description (String)
    - pets accepted (Boolean)
- *comfort*
    - has bed for baby (Boolean)
    - has balcony (Boolean)
    - has TV (Boolean)
    - has yard (Boolean)
    - has fireplace (Boolean)
    - has parking (Boolean)
- *kitchen*
    - has kitchen appliances (Boolean)
    - has dishwasher (Boolean)
    - has freezer (Boolean)
    - has microwave oven (Boolean)
    - has coffeemaker (Boolean)
    - has laundry machine (Boolean)
- *placement*
    - is located outside of town (Boolean)
    - distance to downtown (distance type)
    - is close to oacen (Boolean)
    - is close to beach (Boolean)
    - is close to lake (Boolean)
    - is close to river (Boolean)
    - is close to mountains (Boolean)
    - is close to tourist-attraction (Boolean)
    - is close to restaurant (Boolean)
- *sports*
    - has solarium (Boolean)
    - has whirlpool (Boolean)
    - has pool table (Boolean)
    - has bicycles (Boolean)
    - has tennis court (Boolean)
    - has table tennis (Boolean)
    - has boat (Boolean)
    - has swimming pool (Boolean)
    - has sauna (Boolean)
    - has dart board (Boolean)

B. Selected items of the QUIS applied in the Evaluation

The following instruction was presented at the end of the interaction:

You can help us improve CASTLE by answering the following questions:

Overall Reactions to the software

	0	1	2	3	4	5	6	7	8	9	
terrible	○	○	○	○	○	○	○	○	○	○	wonderful
difficult	○	○	○	○	○	○	○	○	○	○	easy
inadequate power	○	○	○	○	○	○	○	○	○	○	adequate power
dull	○	○	○	○	○	○	○	○	○	○	stimulating
rigid	○	○	○	○	○	○	○	○	○	○	flexible

Screen

Organization of information on screen

confusing	○	○	○	○	○	○	○	○	○	○	very clear

Sequence of screens

confusing	○	○	○	○	○	○	○	○	○	○	very clear

Terminology and system information

Use of terms throughout system

inconsistent	○	○	○	○	○	○	○	○	○	○	consistent

Posistion of message on the screen

inconsistent	○	○	○	○	○	○	○	○	○	○	consistent

Learning

Learning to operate the system

difficult	○	○	○	○	○	○	○	○	○	○	easy

Exploring new features by trial and error

difficult	○	○	○	○	○	○	○	○	○	○	easy

Finding a suitable flat can be performed in a straight-forward manner

true	○	○	○	○	○	○	○	○	○	○	false

Help messages on the screen

unhelpful	○	○	○	○	○	○	○	○	○	○	helpful

System capabilities

System speed

too slow	○	○	○	○	○	○	○	○	○	○	fast enough

System reliability

unreliable	○	○	○	○	○	○	○	○	○	○	reliable

Result

The vacation home that I chose has what I was looking for

true	○	○	○	○	○	○	○	○	○	○	false

I think there is a more suitable vacation home in the catalogue

true	○	○	○	○	○	○	○	○	○	○	false

Bibliography

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communications*, *7*(1), 39–59.

Atkinson, J. W. (1958). *An introduction to motivation.* New York: Van Nostrand Reinhold.

Bergler, R. (1987). *Psychologie in Wirtschaft und Gesellschaft: Defizite, Diagnosen, Orientierungshilfen.* Köln: Dt. Instituts-Verlag.

Bergmann, R. (1998). On the use of taxonomies for representing case features and local similarity measures. In L. Gierl & M. Lenz (Eds.), *Proceedings of the 6th German Workshop on Case-Based Reasoning (GWCBR'98).*

Bräuer, M., & Zimmermann, V. (1997). Fallbasierte Informationssysteme, eine Technologie zur Optimierung der Kunden-Lieferanten-Kommunikation im Internet. In Verein deutscher Ingenieure - Entwicklung, Konstruktion, Vertrieb (Ed.), *Tagungsband: Der Ingenieur im Internet - Neue Chancen durch Engineering-Netze.* Karlruhe.

CBR-Works Frequently Asked Questions. (1998, September). Kaiserslautern.

Chin, J. P., Diehl, V., & Norman, K. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. In *CHI'88 Conference Proceedings: Human Factors in Computing Systems* (pp. 213–218). New York.

Forschungsgemeinschaft Urlaub und Reisen e.V. (FUR). (1997). *RA 96 Kurzfassung; Die Reiseanalyse.* Hamburg.

Giegler, H. (1982). *Dimensionen und Determinanten der Freizeit. Eine Bestandsaufnahme der sozialwissenschaftlichen Freizeitforschung.* Opladen.

Harper, B. D., & Norman, K. L. (1993). Improving user satisfaction: The questionnaire for user satisfaction interaction version 5.5. In *Proceedings of the 1st Annual Mid-Atlantic Human Factors Conference* (pp. 224–228). Virginia Beach, VA.

Hurley, R. F., & Estelami, H. (1998). Alternative indexes for monitoring customer perceptions of service quality. *Journal of the Academy of Marketing Science*, *26*(3), 209–221.

Jameson, A., Schäfer, R., Simons, J., & Weis, T. (1995). Adaptive provision of evaluation-oriented information: Tasks and techniques. In C. S. Mellish (Ed.), *Proceedings of the 14th International Joint Conference on Artifical Intelligence (IJCAI)* (pp. 1886–1893). San Mateo, CA: Morgan Kaufmann.

Kanthak, J. (1973). Möglichkeiten der Marktsegmentierung durch Ermittlung von Urlaubertypen. In *Urlaubsreisen 1972. Bericht über die Auswertungstagung der Arbeitsgemeinschaft Reiseanalyse* (pp. 49–57). Starnberg: Studienkreis für Tourismus. (unpublished study)

Kolodner, J. L. (1993). *Case-based reasoning.* San Mateo: Morgan Kaufmann.

Kosche, G. (1998). E-Commerce: Boom in Sicht. *Markt & Technik*(19), 1.

Kroeber-Riel, W., & Weinberg, P. (1996). *Konsumentenverhalten* (6 ed.). München: Vahlen.

Lenz, M., Auriol, E., & Manago, M. (1998). Diagnosis and decision support. In M. Lenz, B. Bartsch-Spörl, H.-D. Burkhard, & S. Wess (Eds.), *Case-based reasoning technology; from foundations to applications* (pp. 51–90). Berlin: Springer.

Linden, G., Hanks, S., & Lesh, N. (1997). Interactive assessment of user preference models: The Automated Travel Assistant. In A. Jameson, C. Paris, & C. Tasso (Eds.), *User Modeling: Proceedings of the Sixth International Conference, UM97* (pp. 67–78). Vienna, New York: Springer. (http://um.org)

Lödel, D. (1994). *Produktberatung in einem Angebotssystem unter besonderer Berücksichtigung der Kundentypologie.* PhD dissertation, Friedrich-Alexander-Universität Erlangen-Nürnberg.

Lohmann, M., & Besel, K. (1990). *Urlaubsreisen 1989. Kurzfassung der Reiseanalyse 1989.* Starnberg: Studienkreis für Tourismus.

Lohmann, M., & Wohlmann, R. (1987). *Urlaub in Deutschland.* Starnberg: Studienkreis für Tourismus.

McCarthy, J. (1995). Applied experimental psychology. In A. F. Monk & G. N. Gilbert (Eds.), *Perspectives on HCI – diversive approaches* (pp. 75–96). New York: Academic Press.

Mehlmann, O., Landvogt, J., Jameson, A., Rist, T., & Schäfer, R. (1998). Einsatz Bayes'scher Netze zur Identifikation von Kundenwünschen im Internet. *Künstliche Intelligenz*, *12*(3), 43–48.

Meyers, D. G. (1993). *Social psychology* (4 ed.). New York: McGraw-Hill.

Rich, E. (1983). Users are individuals: Individualizing user models. *International Journal of Man-Machine-Studies*, *18*, 199–214.

Richter, M. M. (1998). Indroduction. In M. Lenz, B. Bartsch-Spörl, H.-D. Burkhard, & S. Wess (Eds.), *Case-based reasoning technology; from foundations to applications* (pp. 91–113). Berlin: Springer.

Riesbeck, C. K., & Schank, R. C. (1989). *Inside Case-Based reasoning.* New York: Hillsdale.

Rosewitz, M., & Timm, U. J. (1997). Ideen zur Benutzermodellierung in der elektronischen Produktberatung. In R. Schäfer & M. Bauer (Eds.), *ABIS-97, 5. GI Workshop Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen.* Saarbrücken.

Rössel, M. (1997). Elektronische Branchenkataloge – Akzeptanzförderung durch Benutzermodellierung. In R. Schäfer & M. Bauer (Eds.), *ABIS-97, 5. GI Workshop Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen.*

Rössel, M. (1998). Pragmatische Benutzermodellierung im Adaptiven Multimedialen Präsentationssystem AMPreS. In U. J. Timm & M. Rössel (Eds.), *ABIS-98: 6. Workshop "Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen"* (pp. 67–75). Erlangen.

Schank, R. C. (1986). *Explanation patterns: Understanding mechanically and creatively.* Hilldale: Lawrence Erlbaum Associates.

Siebert, H. (1980). Marketing für die Ewachsenenbildung. Hannover.

Slaughter, L. A., Harper, B. D., & Norman, K. L. (1994). Assessing the equivalence of paper and online versions of the QUIS 5.5. In *Proceedings of the 2nd Annual Mid-Atlantic Human Factors Conference* (pp. 87–91). Washington, D.C.

Smyth, B., & McKenna, E. (1998). Modelling the competence of case-bases. In B. Smyth & P. Cunningham (Eds.), *Proceedings of the 4th European Workshop on Case-Based Reasoning (EWCBR-98): Advances in Case-Based Reasoning* (pp. 208–220). Berlin: Springer.

Sommer, R. (1974). Urlaubsreisen 1972 – Psychologische Segmentationsstudie. Starnberg: Studienkreis für Tourismus. (unpublished study)

Strachan, L., Anderson, J., Sneesby, M., & Evans, M. (1997). Pragmatic user modelling in a commercial software system. In A. Jameson, C. Paris, & C. Tasso (Eds.), *User Modeling: Proceedings of the Sixth International Conference, UM97* (pp. 189–200). Vienna, New York: Springer.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, *59*, 433-450.

Ulich, E. (1994). *Arbeitspsychologie.* Stuttgart: Poeschel.

Weis, T. (1994). *VIPER: Ein verteilter inkrementeller Dialogplaner für eine Multi-Agenten-Umgebung.* Unpublished master's thesis, Department of Computer Science, University of Saarbrücken.

Wilke, W., Lenz, M., & Wess, S. (1998). Intelligent sales support with CBR. In M. Lenz, B. Bartsch-Spörl, H.-D. Burkhard, & S. Wess (Eds.), *Case-based reasoning technology; from foundations to applications* (pp. 91–113). Berlin: Springer.

Erklärung

Hiermit erkläre ich, daß ich die vorliegende Diplomarbeit selbständig verfaßt habe und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Trier, den 23. April 1999